universität wien

MAGISTERARBEIT

Titel der Magisterarbeit

„A comparison of the logistic regression model and the random coefficients linear mixed model for analysis of early kidney disease progression“

Verfasserin

Maria Neubauer, Bakk. rer. soc. oec. Bachelor of Arts

angestrebter akademischer Grad

Magistra der Sozial- und Wirtschaftswissenschaften
(Mag. rer. soc. oec.)

Wien, Oktober 2014

Studienkennzahl lt. Studienblatt:	A 066 951
Studienrichtung lt. Studienblatt:	Magisterstudium Statistik
Betreuer:	Ao. Univ.-Prof. Mag. Dr. Georg Heinze

Acknowledgements

At the beginning I want to thank the people who were involved in the creation of my Master thesis.

First of all, I want to thank Ao. Univ.-Prof. Mag. Dr. Georg Heinze, who is the advisor of my Master thesis. What started as an internship at the Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS) transformed into this Master thesis. I am grateful for this opportunity. Furthermore want to thank him for all his advices, support and patience. I am very grateful to have him as my advisor. I could not think of a better one.

Dr. Milan Hronsky helped me with the simulation. He showed me how to use a simulation computer in the most efficient way. For example, he demonstrated to me how to write .cmd files to run the R Code on all four cores of the simulation computer in an efficient way.

Ass.-Prof. Mag. Dr. Daniela Dunkler had access to the ONTARGET-SysKid dataset, which was analyzed in this Master thesis. I prepared R Code and she ran it on the ONTARGET-SysKid dataset, corrected small code mistakes and sent me the results.

Mag. Maria Kohl introduced me to the program LateX. Furthermore, she was always there for me when I had a question concerning R.

I also want to thank my boyfriend, Mathias Pichler, Bakk., for useful programming tips, last-minute proofreading and his support.

Adrijana Balatinac, BA, did a wonderful work by correcting my spelling and grammar mistakes and contributed therefore immensely to the readability of my Master thesis.

I thank my parents, Felix and Maria Neubauer, and my grandma, Johanna Traunwieser, for the financial support of my Magister study.

And finally I thank my relatives and friends who supported me throughout the entire process.

Many thanks to all of you for your helpful contributions and your believe in me!

Abstract

Chronic kidney disease (CKD) is a major public health problem and therefore its early detection is crucial. Urine albumin to creatinine ratio (UACR) is a continuous clinical marker for the progression (= incidence or progression) of CKD. Consequently, there are two possible approaches to estimate the probability of progression for a patient: either by a logistic approach, using a binary outcome variable defining the status of a patient (progressing or stable), or by a longitudinal approach, using the repeated measurements of UACR as outcome variable. The aim of this Master thesis is to compare these two approaches for the analysis of early kidney disease progression. Therefore the models were compared with regard to their power to detect an association of a risk factor (or treatment) with progression, and with regard to their accuracy in estimating progression rates. It is shown how quantities usually available only with binary outcomes, such as expected relative risks, expected odds ratios, concordance statistics and integrated discrimination improvements can be estimated by either approach, in simulated datasets as well as in a real-life example. All calculations were done using the original log UACR (LOG models) or a Box-Cox log transformation thereof (BCLOG models).

The simulation and the real-life data analysis both showed small but clear differences of the misspecified LOG models compared to BCLOG models. Since the assumption of normally distributed residuals in the longitudinal LOG models was violated, the LOG models underestimated the probability of progression. Therefore, the longitudinal model on Box-Cox transformed log UACR should be the first choice for analysis because of its higher accuracy and unbiasedness.

The longitudinal BCLOG model attained better results than the logistic BCLOG model: the estimated probability of progression by the longitudinal model in the simulation was always closer to the true probability of progression than the probability of progression estimated by the logistic model, which moreover showed a higher variability. The power to detect an effect of a risk factor x on progression was always higher in the longitudinal model than in the logistic model. Testing different scenarios showed that the choice of sample size is crucial for achieving adequate power. Assessing model improvement only lead to small differences in the logistic and longitudinal models.

Furthermore, we noticed a marked difference between apparent progression rates and the true probability of progression, which was due to the fact that patients with an UACR above a certain cut-off point at baseline were excluded from this observational study. Therefore, if UACR is measured with error, then the assessment of progression by categorization of this underlying continuous marker is biased towards higher progression rates. The longitudinal model explicitly estimates this measurement error and therefore leads to approximately unbiased progression rates.

Zusammenfassung

Die Früherkennung der chronischen Nierenerkrankung (engl. CKD), eines der größten aktuellen Gesundheitsprobleme, ist essentiell. Die Albuminausscheidungsrate durch den Urin (engl. UACR) ist ein kontinuierlicher klinischer Marker für Progression der CKD. Folglich gibt es zwei mögliche Ansätze um die Progressionswahrscheinlichkeit eines Patienten/einer Patientin zu schätzen: entweder durch eine logistische Regression, in der eine binäre abhängige Variable verwendet wird, welche den Status eines Patienten/einer Patientin (progredient oder stabil) angibt; oder durch eine longitudinale Herangehensweise, in welcher die wiederholten Messungen von UACR als abhängige Variable verwendet werden. Das Ziel dieser Magisterarbeit ist nun diese beiden Modelle, die logistische Regression und das longitudinale random coefficients model, in der Analyse der Progression der CKD zu vergleichen. Die Modelle werden dabei in Bezug auf ihre statistische Mächtigkeit, den Effekt eines Risikofaktors (oder einer Behandlung) zu sichern, und auf ihre Genauigkeit in der Schätzung von Progressionsraten verglichen. Es wird gezeigt, wie statistische Effektstärkenschätzer, die üblicherweise nur für binäre abhängige Variablen zur Verfügung stehen, wie z. B. erwartetes relatives Risiko, erwartete Odds Ratio, concordance statistics und integrated discrimination improvements, in beiden Ansätzen anhand simulierter Datensätze und eines Praxisbeispiels geschätzt werden können. Alle Berechnungen werden in zwei verschiedenen Situationen durchgeführt: unter Verwendung von log UACR (LOG Modelle) und nach einer Box-Cox Transformation von log UACR (BCLOG Modelle).

Die Simulation und die Analyse des ONTARGET-SysKid Datensatzes haben beide klare Unterschiede zwischen den LOG Modellen und den BCLOG Modellen gezeigt. Da die Annahme normalverteilter Residuen in den longitudinalen LOG Modellen verletzt war, unterschätzten diese Modelle die Progressionswahrscheinlichkeit. Aufgrund ihrer höheren Genauigkeit und Unverzerrtheit sollten die longitudinalen BCLOG Modelle die erste Wahl in die Analyse sein.

Die longitudinalen BCLOG Modelle erreichten bessere Ergebnisse als die logistischen BCLOG Modelle: In der Simulation war die geschätzte Progressionswahrscheinlichkeit in den longitudinalen Modellen immer näher an der wahren Progressionswahrscheinlichkeit als die der logistischen Modelle, welche zudem eine höhere Variabilität aufwiesen. Die Power des Risikofaktors für Progression, welche stark von Stichprobenumfang abhängt, war in den longitudinalen Modellen immer höher als in den logistischen. Die Analyse der Modellverbesserung zeigte nur kleine Unterschiede zwischen den logistischen und longitudinalen Modellen.

Wir fanden einen großen Unterschied zwischen den tatsächlichen Progressionsraten und der wahren Progressionswahrscheinlichkeit, was daran liegt, dass PatientInnen mit einer UACR Messung über

33.9 mg/mmol bei Baseline von der Studie ausgeschlossen wurden. Wenn bei UACR ein Messfehler passiert, dann ist die Kategorisierung des kontinuierlichen Markers in Progression/nicht Progression verzerrt in Richtung höherer Progressionsraten. Das longitudinale Modell schätzt diesen Messfehler explizit und garantiert somit fast unverzerrte Progressionsraten.

Table of Contents

List of figures[1]

[1] POP = probability of progression

List of tables

1 Introduction

Consider a continuous clinical marker which is measured at different time points and consider further a binary risk factor which may have an influence on the clinical marker and thus also on the development of a certain disease. Suppose that a specific cut-off point on this clinical marker defines the presence of a that disease, e.g. if the marker is below the cut-off point, the patient is healthy and if the marker is above the cut-off point, then the patient has developed the disease. Now, what one could be interested in is the probability of the development of the disease over time in patients exposed and not exposed to the risk factor. There are two different ways to estimate this probability. Hence, the purpose of this Master thesis is to compare the performance of these two statistical models: the logistic regression considering a dichotomized clinical marker as outcome variable and the random coefficients linear mixed model for a continuous clinical marker as outcome variable. So, the question is: Is it reasonable to dichotomize a repeatedly measured clinical marker instead of using all information in the model?

Using continuous outcomes is typically more powerful than using dichotomous outcomes, but there are situations in which the power of a well-defined dichotomous outcome approaches or even exceeds that of a continuous outcome based on mean change, as described by Anderson (cf. 2007: 7) in the context of simulation studies for rheumatoid arthritis and ankylosing spondylitis patients. Moreover, Peacock et al (2012) proposed a dual approach which analyses continuous data using means and proportions instead of dichotomization alone and they found out that it is possible (given certain reasonable assumptions) to present dichotomized continuous data and still retain statistical power and precision. Furthermore, the logistic regression approach is often favored for its simplicity and its ability to combine several binary endpoints into one outcome variable.

However, a continuous outcome variable has the advantage to provide more information, as patients with different measurements of the clinical marker are not collapsed into one outcome category. This could be exploited by suitable statistical procedures, such as longitudinal models, that use the continuous clinical marker (or a transformation thereof) as a repeatedly assessed outcome. Further advantages of linear mixed models are that they do not require equally spaced time intervals between consecutive measurements or the same number of measurements per patient and that they can handle nonlinear mean trajectories over time of the clinical marker (cf. Boucquemont et al 2014).

In order to concretize this problem and to establish the opportunity to compare these two approaches, diabetic patients with chronic kidney disease (CKD) have been chosen as our study population and the disease of interest is worsening of CKD. To be precise, this thesis deals with the identification of risk

factors for incidence or progression of chronic kidney disease in diabetic patients, comparing the suitability of these two different statistical models for this purpose. Incidence and progression indicate negative developments of CKD, which are described in detail in the next subsection. The logistic and longitudinal models are compared in regard to their statistical power to evaluate effects of a risk factor of CKD and how accurately model-based progression rates, expected relative risks (RR), expected odds ratios (OR), concordance(c-) statistics and integrated discrimination improvements (IDI) can be estimated by either approach.

The study design and data structure of a real-life study on risk factors for CKD, the 'ONTARGET-SysKid' study (Dunkler et al 2013), are adopted for simulation, and their real data are re-analyzed. ONTARGET is short for ONgoing Telmisartan Alone and in combination with Ramipril Global Endpoint Trial (Mann et al 2008) and SysKid is the abbreviation of SYStems biology towards novel chronic KIDney disease diagnosis and treatment (cf. SysKid 2013). More precisely, SysKid is a large-scale integrating European research project, which aims at deepening our understanding of chronic kidney disease in order to make progress in prevention and to find new diagnostic strategies and treatment options for declining kidney function, which affects millions of patients suffering from diabetes and hypertension (cf. SysKid 2013). In the ONTARGET study the effects of telmisartan (an angiotensin receptor blocker), ramipril (an angiotensin converting enzyme) and their combination, used at full doses, on renal outcomes were examined in a large population at high cardiovascular risk as described by Mann et al (2008). The randomized, double-blind and controlled trial recruited 25,620 participants who were aged 55 years or older and had developed atherosclerotic vascular disease or diabetes with end-organ damage, and followed them up from 2001 to 2007 (cf. Mann et al 2008). In this Master thesis we focus on diabetic patients, a subpopulation of the ONTARGET dataset, in the sequel denoted by the 'ONTARGET-SysKid' subpopulation. To be precise, 5,293 individuals with type 2 diabetes with normo- or microalbuminuria (i.e. either no kidney problems or beginning excretion of proteins) at baseline from the ONTARGET dataset were included in this observational study, which is the same setting as in the ONTARGET-SysKid study as described by Dunkler et al (2013). Patients were recruited from January 2002 to July 2003, with a prospective follow-up through January 2008, and the clinical marker of interest, i.e. urine albumin to creatinine ratio (UACR), was measured at baseline, at two and at five years from randomization.

Since UACR is a continuous clinical marker, which is used to indicate incidence or progression of chronic kidney disease, we can explore the two different approaches from above on this dataset. In order to describe incidence or progression of chronic kidney disease in a patient, we can either treat incidence or progression as a binary variable (yes/no at 5 years, if UACR crosses a threshold value that defines micro-

or macroalbuminuria) or as a longitudinal continuous process (by the UACR measurement or a transformation thereof). Thus, the logistic model takes the binary outcome information (progressing or stable) at a particular time point (e.g., 5 years) as outcome variable and does not require any transformation of this outcome variable. This model supplies estimates of odds ratios of risk factors, which measure the strength of association of a risk factor, possibly adjusted for confounders, with the progression status of a patient. In addition, this model directly supplies model-based estimates of progression rates for patients with various values of risk factors and confounders at baseline. These estimates can then further be used to compute c-statistics, and to quantify the added value of a risk factor in terms of the integrated discrimination improvement.

The longitudinal model takes all available repeatedly measured values of UACR as outcome variable, and fits a general linear mixed model with random coefficients. To fulfill assumptions on the distribution of residuals, UACR have to be transformed before statistical analysis, which affects the interpretation of the parameter estimates. Because of their very skew distribution, a log transformation may be first choice to achieve normally distributed residuals. We will also consider a subsequent Box-Cox transformation (cf. Box and Cox 1964) of the log UACR. In order to demonstrate the beneficial effect of the Box-Cox transformation, we will perform all calculations using the Box-Cox transformation on the log-transformed UACR (BCLOG model) and using log-transformed UACR, without further transformation (LOG model). We will show that the parameters of a suitable Box-Cox-transformation can be found by optimizing the agreement of residuals with a normal distribution. We will further show how the results from such a longitudinal model can be back-transformed into predicted progression rates, allowing the expression of the estimated effects as expected odds ratios, c-statistics and IDI, such that a direct comparison with the logistic model is possible.

The remainder of this thesis is organized as follows. Section 2 provides a short introduction to the most important medical terms which are used in this Master thesis. In section 3, the methods used for the statistical assessment of risk factors for incidence or progression are explained. Section 4 concentrates on the data analysis. This section is split into two parts: a simulation study and the analysis of the ONTARGET-SysKid study. Finally, the thesis closes with a conclusion, providing a discussion about the investigated methods and stating the limitations of this exploratory study. Data analysis was implemented using several self-written R functions explained in the appendix.

All calculations and visualizations were conducted using the statistical software R version 2.13.0 (R Development Core Team 2011).

2 Medical Definitions

2.1 Chronic kidney disease

Chronic kidney disease (CKD), a major public health problem, is defined according to the presence or absence of kidney damage and level of kidney function—irrespective of the type of kidney disease (cf. National Kidney Foundation). Generally, CKD can be heterogeneous disorders, which affect the structure and function of the kidney (cf. Boucquemont et al 2014). Usually, CKD is a progressive and hardly reversible disease (cf. Boucquemont et al 2014). "When kidney disease progresses, it may eventually lead to kidney failure, which requires dialysis or a kidney transplant to maintain life" (National Kidney Foundation).

Nowadays, about 10 % of Europe's population are affected of CKD, which are approximately 50 million patients (www.syskid.com). Most of these patients are at an early phase of CKD, in which therapeutic actions still have a positive impact and can prevent progression or worse (www.syskid.com) Therefore, an early detection of CKD is crucial (cf. National Kidney Foundation). Symptoms of CKD include: feeling more tired and having less energy, having trouble concentrating, having a poor appetite, having trouble sleeping, having swollen feet and ankles, having dry, itchy skin, etc. (cf. National Kidney Foundation).

The main causes of chronic kidney disease are diabetes and hypertension (cf. National Kidney Foundation). Diabetes occurs when the blood sugar is too high, which causes damage to many organs including the kidneys, heart, blood vessels, nerves and eyes (cf. National Kidney Foundation). Hypertension happens when the pressure of the blood against the walls of the blood vessels increases, which may lead to a heart attack, stroke and chronic kidney disease (cf. National Kidney Foundation). Interestingly, chronic kidney disease can not only be caused by high blood pressure, but it can also be the cause of high blood pressure (cf. National Kidney Foundation).

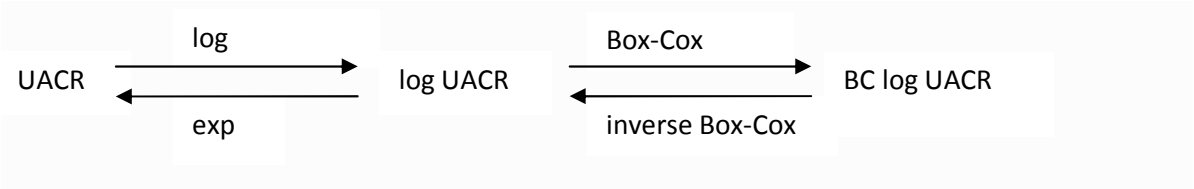
2.2 Urine albumin to creatinine ratio

Urine albumin excretion (or urine albumin to creatinine ratio; short: UACR) is a continuous clinical marker for chronic kidney disease (cf. National Kidney Disease Education Program). UACR is the ratio of urine albumin and urine creatinine and it estimates the 24-hour urine albumin excretion (cf. National Kidney Disease Education Program). Urine albumin excretion can be either measured in mg/ml or mg/mmol: 1 mg/mmol equals 8.849558 mg/ml. In the following context we work with the unit mg/mmol. Clinically, urine albumin excretion is categorized into three intervals: 0 to 3.39, 3.40 to 33.9 and above 33.9 mg/mmol as described by Dunkler et al (2013). The first category is called normoalbuminuria, indicating no kidney problems, the second category microalbuminuria (beginning

excretion of proteins) and the third category macroalbuminuria (progressive protein excretion) (cf. Dunkler et al 2013). Stable or lower levels of albuminuria may indicate that therapy is effective and increasing levels may indicate a worsening of CKD (cf. National Kidney Disease Education Program). Effective therapies might be decreasing excessive dietary protein and sodium to healthful levels and the use of angiotensin converting enzyme (ACE) inhibitors and angiotensin receptor blockers (ARB) (cf. National Kidney Disease Education Program).
Furthermore, urine albumin excretion is an early predictor of cardiovascular disease morbidity and mortality and renal function loss (cf. National Kidney Disease Education Program).

As outlined in the introduction, we will not only work with untransformed UACR, but also with their log transformations (log UACR) and with further transformed values, i.e. applying a Box-Cox-transformation to the log UACR (BC log UACR). For the log transformation the natural logarithm was used. Figure 1 gives a short overview of the here applied UACR transformations which were applied in this Master thesis.

Figure 1. Overview of UACR transformations



We have written the R functions `boxcox` and its inverse function `invboxcox` to transform the log UACR by applying optimal Box-Cox parameters. The definition of the optimal Box-Cox parameters as well as how they can be found, is explained in subsection 4.1.

In order to get a clearer picture of the transformations, we simulated 5000 baseline UACR measurements (details on how the UACR measurements are simulated will follow in subsection 4.2) and looked at their relationship to the log and Box-Cox log transformation, respectively. Figure 2 illustrates the exponential relationship between the UACR measurements and their log transformation. Additionally, the distribution of the 5000 log transformed UACR measurements (top panel) and the distribution of the 5000 UACR measurements (right panel) are visualized through histograms. For example, an UACR of 10 mg/mmol represents a log UACR of 2.3.

Figure 2. UACR and the log transformation thereof; n = 5000

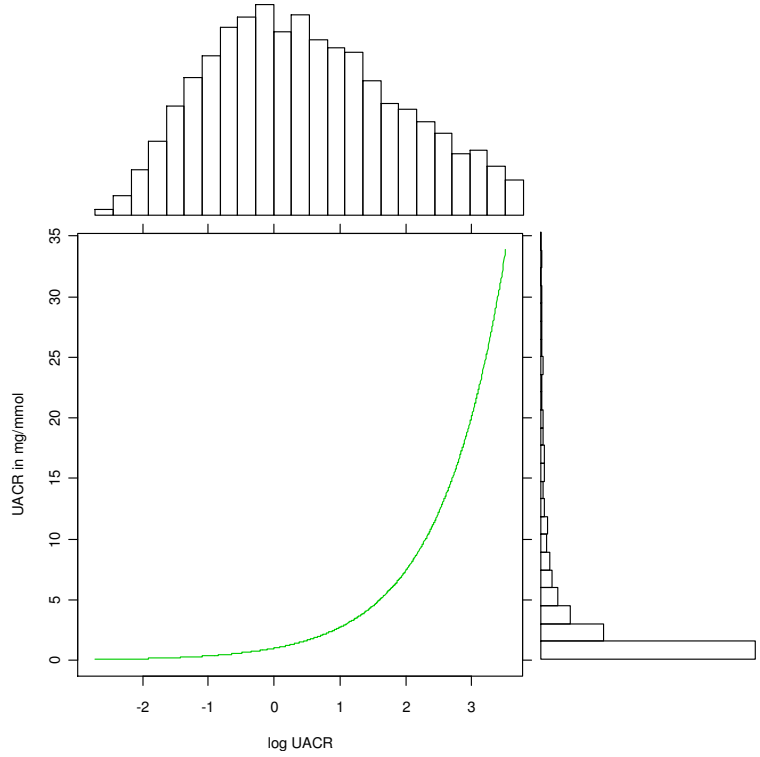
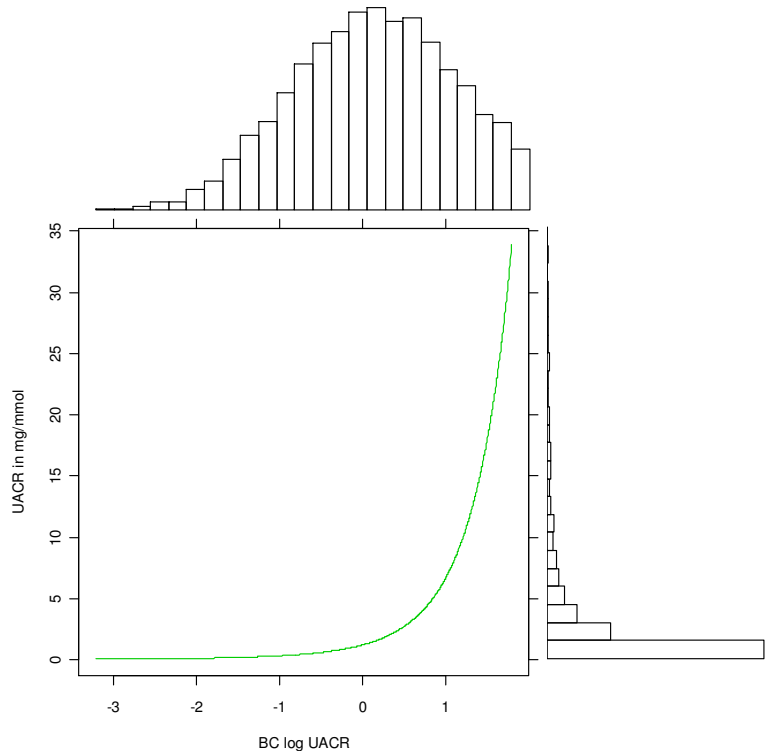


Figure 3 shows how the urine albumin excretion values are related to the Box-Cox log transformation thereof. Like before, their corresponding distributions are visualized by histograms. For example, an UACR of 10 mg/mmol represents a BC log UACR of 1.2.

Figure 3. UACR and the Box-Cox log transformation thereof; n = 5000



2.3 Incidence and Progression

In this thesis, incidence of kidney disease is defined as a change from normoalbuminuria to micro- or macroalbuminuria, while progression is defined as a change from micro- to macroalbuminuria within a particular time frame. In the remainder of this thesis, we will use the notation 'progression' to denote incidence or progression of kidney disease.

In the ONTARGET study, albuminuria was assessed two and five years after the inclusion of a patient, and thus progression can be evaluated at these two time points. To guard against random fluctuations of albuminuria an additional condition was introduced, requiring an increase of at least 30 % compared to the baseline urine albumin excretion value to define a progression cut-off point as described by Dunkler et al (2013). This definition is implemented in the calculation of progression by the self-written R functions `cutpoint` and `progression` (see R functions in the appendix). In the following, progression is only considered at the time point five years after inclusion.

Table 1 summarizes the definitions of incidence or progression of chronic kidney disease as described by Dunkler et al (2013). If a patient presents with normoalbuminuria at baseline, his or her individual cut-off point for defining incidence of CKD at follow-up is defined as either the maximum of 3.39 mg/mmol and 1.3 times the baseline value for a transition to microalbuminuria or as the maximum of 33.9 mg/mmol and 1.3 times the baseline value for a transition to macroalbuminuria. If a patient presents with microalbuminuria already at baseline, his or her individual cut-off point for defining progression of CKD at follow-up is defined as 33.9 mg/mmol or 1.3 times the baseline value, whichever is higher. Patients with macroalbuminuria at baseline are not included in the ONTARGET-SysKid study.

Table 1. Overview of the definitions of incidence and progression

Transition in albuminuria	Baseline value of UACR	Follow-up value of UACR
normo → micro (incidence)	<3.39 mg/mmol	>max(3.39, 1.3 times baseline UACR)
normo → macro (incidence)	<3.39 mg/mmol	>max(33.9, 1.3 times baseline UACR)
micro → macro (progression)	<33.9 mg/mmol	>max(33.9, 1.3 times baseline UACR)

Table 2 summarizes the 'hard' cut-off points for progression on three scales, i.e. the cut-off points without the 30 % rule.

Table 2. Summary of all cut points for progression

Cut-off points	UACR*	Log UACR	BC log UACR
first (incidence)	3.39	1.22	0.43
second (incidence/progression)	33.9	3.52	1.65

* measured in mg/mmol

3 Methods

This section introduces the methods which were used in this Master thesis. First, some notation conventions are explained (see subsection 3.1). Then, the statistical assessment of risk factors for progression is discussed in general (see subsection 3.2). In order to concretize our problem, the methods used for the data analysis are described in the subsections simulation (see subsection 3.3) and analysis of the ONTARGET-SysKid study (see subsection 3.4). A suitable transformation for the UACR is proposed in subsection 3.5. Finally, subsection 3.6 deals with the assessment of model improvement.

3.1 Notation

We consider a dataset on n individuals, with Y_{ti} denoting the urine albumine creatinine ratio measured in individual i ($i = 1,\ldots,n$) at time point t, and X_i denoting the value of a risk factor at baseline. In accordance with the setting of the ONTARGET-SysKid study, we consider $t = 0$, 2 and 5 years and for simplicity, we assume in the simulation that X is a binary risk factor, e.g., smoking status at baseline, assuming values of 0 or 1. At baseline, all individuals are either normo- or microalbuminuric, which corresponds to values of Y_{0i} lower than 3.39 mg/mmol or between 3.39 and 33.9 mg/mmol, respectively. In the ONTARGET-SysKid study, patients presenting with macroalbuminuria (Y_{0i} > 33.9 mg/mmol) at baseline were excluded.

Let A_{ti} denote a binary indicator of progression (1 denoting incidence or progression, 0 denoting no change or improvement in albuminuria class) for individual i at time point t, compared to the baseline status. Here we only consider progression at 5 years, which we write as $\mathrm{A_i} \coloneqq \mathrm{A_{5i}}$.

As in subsection 2.1 already introduced, we will work in the following with two different kinds of urine albumin excretion value transformations. Models using log UACR as input variable are denoted by 'LOG model' and models using Box-Cox transformed log UACR as input variable, in order to meet the assumption of normally distributed residuals in the longitudinal models, are indicated by the label 'BCLOG model'.

3.2 Statistical assessment of risk factors for progression

There are two principles to analyze data as defined above if the analysis task is to evaluate risk factors in their effect on the progression of kidney disease. The first one uses logistic regression to model the probability of progression, expressed by the outcome variable A, as a function of X and covariates Z_2 to Z_k. By contrast, the second approach makes use of the repeated measurements on the continuous outcome variable Y, possibly after a suitable transformation, and applies a linear mixed model with random coefficients to evaluate the relationship with X and Z_2 to Z_k.

By the logistic regression model in notation following Harrell (2001), the probability of progression, i.e., of $A_i = 1$, is given by

$$\Pr(A_i = 1) = \frac{1}{1+\exp(-(\alpha_0+\alpha_1 X_i+\alpha_2 Z_2+\cdots+\alpha_k Z_k))} \quad (1)$$

The odds ratio for progression comparing an individual exposed to the risk factor X ($X_i = 1$) with a non-exposed individual ($X_i = 0$) is given by $\exp(\alpha_1)$. The probability of progression for exposed and non-exposed subjects adjusted for $Z_2, \ldots, Z_k$ can be derived by inserting sample means for $Z_2, \ldots, Z_k$ into (1), and setting X to 1 or to 0, respectively. Since a progression in albuminuria class is more likely if the baseline value is already close to a cut-off point defining an albuminuria class change, it may be useful to consider as covariate the difference of the baseline value and the cut-off point on the log scale (cf. Dunkler et al 2013).

A regression model considering the repeated measurements Y_{ti} of the continuous outcome variable could be defined in notation following Brown and Prescott (1999) as

$$Y_{ti} = \beta_0 + \beta_1 X_i + \beta_2 t + \beta_3 t X_i + b_{0i} + b_{1i} t + \varepsilon_{ti} \quad (2)$$

$\varepsilon_{ti} \sim N(0, {\sigma_\varepsilon}^2)$ denotes a normally distributed random error for individual i at time t with mean 0 and variance ${\sigma_\varepsilon}^2$. The random coefficients of individual i are denoted by b_{0i} and b_{1i} (intercept and slope, respectively) and are assumed to follow a bivariate normal distribution with mean values of 0 and covariance matrix G. The diagonal elements of G are denoted by σ_0^2 and σ_1^2, and the off-diagonal as σ_{01}^2. This model assumes that the expected value of the outcome variable Y at time point 0 differs for exposed and non-exposed individuals by β_1, and at time point t by $\beta_1 + t\beta_3$. If the risk factor X is not correlated with baseline values Y_0, then $\beta_1 = 0$. In case that exposure to the risk factor X accelerates the increase of UACR compared to non-exposure, then β_3 will be positive. Thus, given equal baseline values of UACR in exposed and non-exposed subjects, a non-zero β_3 coefficient indicates an association of the risk factor with a more rapid increase in UACR. b_{0i} and b_{1i} are individual deviations from the average UACR trajectories defined by β_0, β_1, β_2, and β_3 and the value of the risk factor X_i for an individual i. The linear mixed model with random coefficients does not explicitly estimate b_{0i} and b_{1i} individually, but estimates their covariance matrix G. So-called best linear unbiased predictors (BLUP) of b_{0i} and b_{1i} can be estimated after model fitting by the empirical Bayes method (cf. SAS 2014).

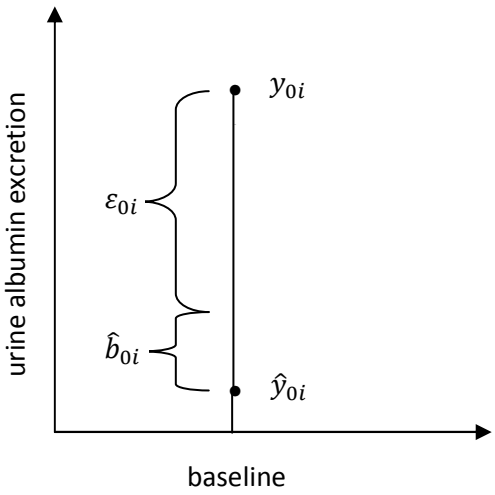
Values of Y for new subjects at time point t can be predicted using (2). If no baseline value is available, the predicted value at time point t for an individual i is given as

$$\hat{y}_{ti} = \beta_0 + \beta_1 X_i + \beta_2 t + \beta_3 t X_i$$

If a baseline value y_{0i} is available, then this information can be incorporated to obtain more precise predictions for y_{2i} and y_{5i}. The random intercept for individual i is estimated by incorporating the empirical Bayes estimate using

$$\hat{b}_{0i} = \sigma_0^2 \frac{y_{0i} - \hat{y}_{0i}}{\sigma_0^2 + \sigma_\varepsilon^2}$$

Figure 4. Difference between estimated and true UACR at baseline



Here, we assume that the difference between the observed baseline value y_{0i} and $\hat{y}_{0i}$, the value estimated from fixed effects, is a sum of the random intercept of subject i and the error ε_{0i} of subject i at baseline as shown in Figure 4. The total variance of $y_{0i} - \hat{y}_{0i}$ is a sum of the variance of the random effect b_0, σ_0^2, and the variance of the error ε, σ_ε^2. Thus, we attribute the fraction $\frac{\sigma_0^2}{\sigma_0^2+\sigma_\varepsilon^2}$ to the random effect b_0, estimating this random effect for patient i at baseline as $\frac{(y_{0i}-\hat{y}_{0i})\cdot\sigma_0^2}{\sigma_0^2+\sigma_\varepsilon^2}$. An estimate of the random slope of this subject is obtained by the expectation of the random slope b_1 conditional on the random intercept $\hat{b}_{0i}$, making use of the bivariate normal distribution of b_0 and b_1:

$$\hat{b}_{1i} = \frac{\sigma_{01}\hat{b}_{0i}}{\sigma_0^2}$$

Then, updated predictions for that subject at time point t are given by

$$\tilde{y}_{ti} = \hat{y}_{ti} + \hat{b}_{0i} + t\hat{b}_{1i}$$

The variance of the individual prediction of Y at t given baseline value y_{0i} is given by

$$V_{ti} = V_{0i} + (1\ t)G_{0i}\binom{1}{t} + \sigma_\varepsilon^2,$$

where $V_{0i} = x_{0i}' V x_{0i}$ denotes the variance of the mean prediction of Y at time t, given x_{0i} is the vector of fixed covariates entering (2), i.e., $x_{0i} = [1, X_i, t, tX_i]'$, V is the covariance matrix of the

fixed effects parameters $\beta = [\beta_0, \beta_1, \beta_2, \beta_3]'$, and G_{0i} is the conditional covariance matrix of the random effects given the baseline value y_{0i}, with diagonal entries $\sigma^2_{0|0i} = \left(\frac{\sigma_0^2}{\sigma_0^2+\sigma_\varepsilon^2}\right)^2 V_{0i}$ and $\sigma^2_{1|0i} = \left[1 - \left\{\frac{\sigma_{01}}{\sigma_0\sigma_1}\right\}^2\right]\sigma_1^2$, and off-diagonal $\sigma^2_{01|0i} = \frac{\sigma^2_{01}}{\sigma_1^2}$.

If model (2) holds, then the probability of progression at time point t can be obtained by a normal approximation:

$$\Pr(A = 1) = 1 - \Phi\left(\frac{C_i - \tilde{y}_{ti}}{\sqrt{V_{ti}}}\right)$$

where C_i is the patient-individual cut-off point for defining albuminuria incidence ($C_i = \max\{3.39,\ 1.3 \cdot y_{0i}\}$ or $\max\{33.9,\ 1.3 \cdot y_{0i}\}$ mg/mmol if y_{0i} < 3.39 mg/mmol) or progression ($C_i = \max\{33.9,\ 1.3 \cdot y_{0i}\}$ if 3.39 mg/mmol < y_{0i} < 33.9 mg/mmol), and $\Phi(s)$ denotes the standard normal cumulative distribution function at s.

In practice, model (2) will be estimated from the data and all calculations follow straightforwardly by plugging in estimates for the quantities β, G, and σ_ε^2.

Note that in the following, the logistic regression models are often abbreviated as 'logistic models' and the linear mixed models with random coefficients as 'longitudinal models'.

3.3 Simulation

We simulated datasets in order to compare the power of the logistic and the longitudinal model in detecting a significant effect of a single binary risk factor on progression (for the logistic model) or increase of UACR in time (for the longitudinal model). Values of the binary risk factor were drawn from a $Bernoulli(0.5)$ distribution. The sample size for our investigation was chosen such that the power in the logistic approach for detecting an expected relative risk of 1.2 was approximately 80 %. The simulated datasets were generated using a longitudinal BCLOG model and inserting the estimated values of G and σ_ε^2, which were obtained from the ONTARGET-SysKid study. To amplify this, for the UACR measurements of risk group $X = 0$ the intercept and slope were drawn from a bivariate normal distribution for n patients with a mean Box-Cox transformed log UACR of β_0, and β_2, respectively, and a variance-covariance matrix G. A normally distributed error with mean zero and the residual sum of squares (RSQE) from the longitudinal model as standard deviation was added to each measurement. For group $X = 1$ the UACR measurements were simulated analogously (for details see subsection 4.2.1.1). β_0, β_1, and β_2 from model (2) were set to 0. The effect of the binary risk factor was varied by assuming equally-spaced values of β_3 ranging from 0 to 0.05 such that the expected odds ratio for progression ranged from 1 to 1.3. For each β_3 we simulated 1000 datasets of n patients.

In addition, we worked with different scenarios, i.e. the simulation with and without the optimization of the Box-Cox parameters for the BCLOG models and a change in sample size. The first six scenarios follow the setting as described above. Then there are two scenarios in which the sample size was halved for $\beta_3 = 0.05$ and doubled for $\beta_3 = 0.02$. In those scenarios the global Box-Cox parameters (see subsection 4.1) were used. In further two scenarios ($\beta_3 = 0.02$ and $\beta_3 = 0.05$) the Box-Cox parameters were optimized on each of the 1000 simulated datasets.

Recall the model definitions from subsection 3.2. In all scenarios we consider only the binary risk factor X as main covariable, which leads to the model definitions in subsection 3.3.1 and 3.3.2, respectively. The logistic and longitudinal models were evaluated on each data set two times: once with and once without the risk factor X. Finally, we compared the accuracy of the predicted probabilities for progression derived from both models at various baseline values by means of measures of discrimination and predictive accuracy (see subsection 3.6).

3.3.1 Logistic models

When the data was analyzed in the simulation according to the logistic model, the following model formula was assumed:

$$\Pr(A = 1) = \frac{1}{1+\exp\left(-[\alpha_0 + \alpha_1 \cdot x + \alpha_2 \cdot d]\right)},$$

where x is a binary risk factor (e.g. the smoking status of a patient) defined as $x = 1$ if the patient is exposed to the risk factor and $x = 0$ if the patient is not exposed to the risk factor. Two different versions of the logistic model were estimated, the logistic LOG model, using log UACR, and the logistic BCLOG model, which uses Box-Cox-transformed log UACR. This distinction is relevant only for the definition of d. The independent variable d is the difference between the log UACR at baseline and the log cut-off point for declaring progression given that baseline UACR (logistic LOG model), or the difference between the Box-Cox-transformed log UACR at baseline and the Box-Cox-transformed log cut-off point (logistic BCLOG model).

3.3.2 Longitudinal models

When the data was analyzed in the simulation according to the longitudinal model, the following model was assumed:

$$Y_{ti} = \underbrace{\beta_0 + \beta_1 \cdot x + \beta_2 \cdot t + \beta_3 \cdot x \cdot t}_{\text{fixed effects}} + \underbrace{b_{0i} + b_{1i} \cdot t}_{\text{random effects}} + \varepsilon_{ti}$$

$$\varepsilon_{ti} \sim N(0, SQE)$$

$$(b_{0i},\, b_{1i}) \sim MVN(0, G)$$

This representation of the model by the fixed-effects coefficients, β_0, β_1, β_2 and β_3, corresponding to the intercept, the group effect, the time effect and the interaction effect, constitutes an 'interaction parameterization'. In order to express the assumed groupwise trajectories, the so-called 'groupwise parameterization' is more useful. In this parameterization, γ_{00}, γ_{01}, γ_{10} and γ_{11} denote intercept and slope of the trajectories in the two groups with risk factor $x = 0$ or $x = 1$, where γ_{00} and γ_{01} refer to group $x = 0$ and γ_{10} and γ_{11} refer to group $x = 1$. Table 3 summarizes the meaning of the model parameters in both parameterizations.

Table 3. Overview of simulation parameters

Groupwise parameterization	Description		Interaction parameterization
γ_{00}	Intercept	$x = 0$	β_0
γ_{01}	Slope	$x = 0$	β_2
γ_{10}	Intercept	$x = 1$	$\beta_0 + \beta_1$
γ_{11}	Slope	$x = 1$	$\beta_2 + \beta_3$

3.4 Analysis of ONTARGET-SYSKID study

Our investigation was confined to the setting of the ONTARGET-SysKid study as described by Dunkler et al (2013). 6,213 patients from ONTARGET with type 2 diabetes mellitus and with normo- or micro-albuminuria at baseline were followed up for a mean of 5.5 years, and their UACR was measured at baseline, at two and at five years from randomization (cf. Dunkler et al 2013). In this Master thesis only patients with all three UACR measurements were included, resulting in a study population of 5,293 patients. In real analyses, the competing risk of death during follow-up should be considered, but an analysis accounting for death was beyond the scope of this investigation.

We investigated the impact of randomization groups on the risk of progression of chronic kidney disease at five years, using the logistic and the longitudinal regression approaches outlined above.

3.4.1 Logistic models

When the data was analyzed according to the logistic model, the following model formula was assumed:

$$\Pr(A = 1) = \frac{1}{1 + \exp\left(-(\alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 d)\right)}$$

The dummy variables x_1 and x_2 were defined as the randomization groups for the double-blind treatment of the patients. The treatment groups 'Ramipril', 'Telmisartan' and 'Combination' are denoted by coding two binary variables x_1 and x_2 as shown in Table 4. Two different versions of the logistic model were estimated, the logistic LOG model, using log UACR, and the logistic BCLOG mod-

el, which uses Box-Cox-transformed log UACR. This distinction is relevant only for the definition of d. Recall that d is the difference between the log UACR at baseline and the log cut-off point for declaring progression given that baseline UACR (logistic LOG model), or the difference between the Box-Cox-transformed log UACR at baseline and the Box-Cox-transformed log cut-off point (logistic BCLOG model).

Table 4. Randomization groups expressed as dummies

Dummies	**Randomization groups**		
	Ramipril	Combination	Telmisartan
$\boldsymbol{x_1}$	1	0	0
$\boldsymbol{x_2}$	0	1	0

3.4.2 Longitudinal models

When the ONTARGET-SysKid data was analyzed according to the longitudinal model, the following model was assumed:

$$Y_{ti} = \underbrace{\beta_0 + \beta_1 \cdot t + \beta_2 \cdot x_1 + \beta_3 \cdot x_2 + \beta_4 \cdot x_1 \cdot t + \beta_5 \cdot x_2 \cdot t}_{\text{fixed effects}} + \underbrace{b_{0i} + b_{1i} \cdot t}_{\text{random effects}} + \varepsilon_{ti}$$

$$\varepsilon_{ti} \sim N(0, SQE)$$

$$(b_{0i},\ b_{1i}) \sim MVN(0, G)$$

Here the randomization groups are expressed in dummy notation as before (see Table 4). To be precise, this formula was used for producing the longitudinal LOG models and the BCLOG models.

3.5 Transformation of UACR

If the normal assumption is in doubt (as will usually be the case with urine albumin excretion values, but also with their logarithms), we suggest to employ a transformation to the values Y_{ti} such that the empirical distribution of the residuals ε_{ti} of (2) is approximately normal. Therefore we searched for a suitable transformation of the logarithmized urine albumin to creatinine ratio.

In a first step we generated 1000 normally distributed values, where mean and standard deviation were taken from the log UACR of a subsample of the ONTARGET-SysKid dataset. The next step was to find a suitable transformation that maps normally distributed values to a plausible distribution of UACR. To this end, the normally distributed values were taken to the powers from -5 to 5 at intervals

of 0.5. In order to avoid undefined values caused by non-integer power exponents with negative arguments, we first shifted the data into the positive range, and scaled them such that their mean equaled 1 ($y = x - min(x) + 0.01$). We also removed the data in the tails by truncating at the 5thand 95th percentiles. Obtaining non-satisfying results, we tried a different approach.

In order to fulfill the assumption of normally distributed residuals in the longitudinal models we employed a suitable Box-Cox-type (cf. Box and Cox 1964) transformation to the values Y_{ti} such that the empirical distribution of the residuals of (2) was approximately normal. Model (2) can be extended to include other covariates, e.g. for adjustment of the effect of X. With real-life data, we obtained acceptable results by first taking the logarithm of the urine albumin excretion values, and then applying the Box-Cox transformation.

The Box-Cox transformed value E_1 , which was used in the simulation, is given by

$$E_1 = \begin{cases} \dfrac{(\log(U) + c)^{\lambda} - 1}{\lambda}, & \lambda \neq 0 \\ \log(\log(U) + c), & \lambda = 0 \end{cases}$$

where U is the urine albumin excretion value. The transformation parameters λ and c were optimized by minimizing the sum of squared differences of the standardized residuals ε_{ti} in a model of type (2) applied to the ONTARGET-SysKid study including time t as the only fixed covariate, and the corresponding theoretical normal quantiles. Additionally, E_1 was standardized by subtracting its mean and dividing by its standard deviation. In the sequel, these parameters were assumed to be constant in the simulation for computational reasons. Only for two cases, $\beta = [0{,}0{,}0{,}0.02]'$ and $\beta = [0{,}0{,}0{,}0.05]'$, the transformation parameters were optimized on each dataset in 1000 simulation runs, to see whether treating the Box-Cox transformation parameters as fixed has any impact on the accuracy of results.

In the analysis of ONTARGET-SysKid study the outcome variable log UACR in the longitudinal model was first transformed using a Box-Cox transformation with only the parameter λ. The Box-Cox transformed value E_2 is given by

$$E_2 = \begin{cases} \dfrac{(\log(U))^{\lambda} - 1}{\lambda}, & \lambda \neq 0 \\ \log(\log(U)), & \lambda = 0 \end{cases}$$

where U is the urine albumin excretion value. The transformation parameter λ was optimized and E_2 was standardized by subtracting its mean and dividing by its standard deviation. In the sequel, these parameters were considered as constants.

3.6 Assessment of model improvement

"A key measure of the clinical utility of a survival model is its ability to discriminate or separate those who will develop the event of interest from those who will not" (Pencina et al 2008: 158). The area under the receiver-operating-characteristic (ROC) curve is the most popular metric for this kind of discrimination (cf. Pencina et al 2008).

In this Master thesis, concordance statistics, the correlation of the predicted probability of progression and observed progression status, integrated discrimination improvements (IDI) were used for assessing model improvements of the logistic and longitudinal models in the simulated datasets as well as in the ONTARGET-SysKid dataset. In some cases, the net reclassification improvement (NRI) was provided as well. The concordance statistic (short: c, c-statistic or c-index) is a measure of classification accuracy as described by Vittinghoff (2005) and Harrell (2001). The c-statistic is derived from the Wilcoxon-Mann-Whitney two-sample rank sum test and is computed by taking all possible pairs of subjects always including a response and a non-response (Harrell 2001: 247). As pointed out, e.g., by Steyerberg (2009), the c-index is identical to the area under the ROC curve, treating the response status as 'disease status' and the linear predictor of a model as 'disease marker'. A concordance index of 1 denotes perfect prediction, i.e. perfect separation of responders and non-responders, and a c-index of 0.5 symbolizes a random prediction like flipping a coin (Steyerberg 2009: 247).

The IDI evaluates the usefulness of a new marker by focusing on differences between integrated sensitivities (IS) and one minus integrated specificities (IP) for models with (new) and without (old) a new marker as described by Pencina et al (2008). Hence the IDI is defined as:

$$IDI = (IS_{new} - IS_{old}) - (IP_{new} - IP_{old})$$

In the simulation the binary variable X was used as the new marker and in the analysis of the ONTARGET-SysKid dataset the randomization groups were chosen as the new marker.

The NRI, a further measure of discrimination, was used to check the plausibility of the IDI results in the simulation (see subsection 4.2.2.1.2.3). The NRI measures the improvement in reclassification, which "can be quantified as a sum of differences in proportions of individuals moving up minus the proportion moving down for people who develop events, and the proportion of individuals moving down minus the proportion moving up for people who do not develop events." (Pencina et al 2008:160). Therefore, the NRI is defined by Pencina et al (2008: 161) as:

$$NRI = [P(up|A = 1) - P(down|A = 1)] - [P(up|A = 0) - P(down|A = 0)]$$

4 Data analysis

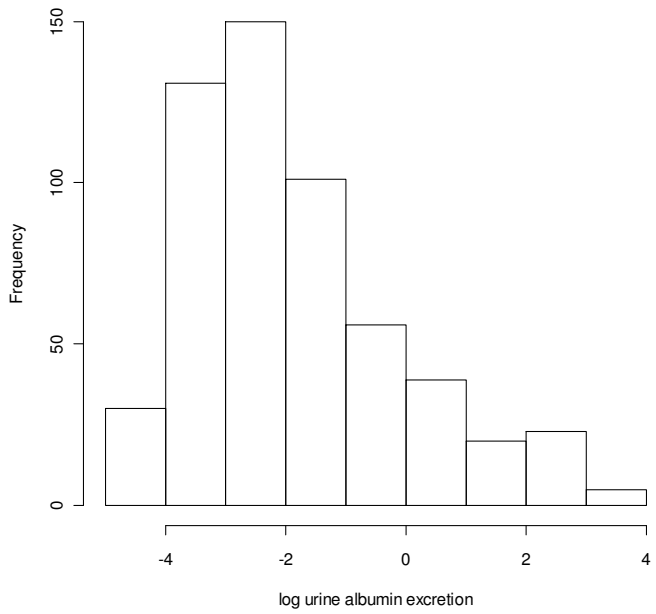
The performance of the logistic approach compared to the performance of the longitudinal approach was analyzed in two different settings: as a simulation study (see subsection 4.2) and by a comparative analysis of the ONTARGET-SysKid dataset (see subsection 4.3). Prior to further analysis, a suitable transformation for the urine albumin to creatinine ratio was searched for, to guarantee the agreement of residuals with a normal distribution in the longitudinal models (see subsection 4.1).

4.1 Searching the best transformation of UACR

Since usually the normal assumption is in doubt with urine albumin excretion values, we searched for an appropriate transformation to the values Y_{it} such that the empirical distribution of the residuals in the longitudinal models is approximately normal.

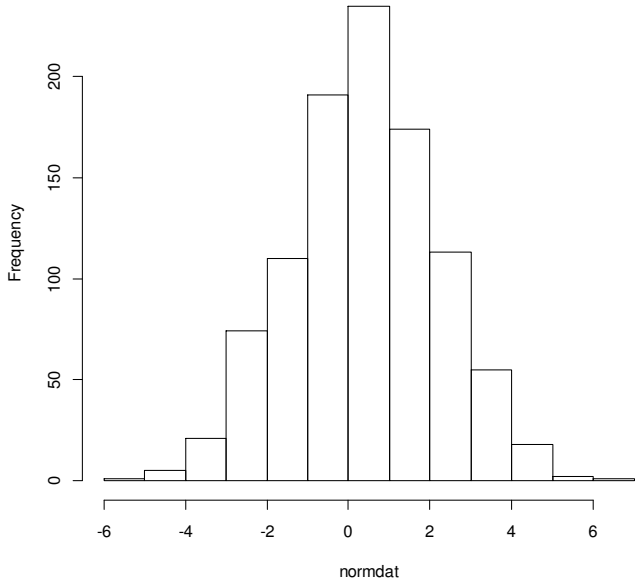
First of all, we visualized the distribution of log UACR. Figure 5 shows the logarithmized urine albumin excretion values of a random subsample of the ONTARGET-SysKid dataset consisting of 200 patients. UACR was measured in all patients at baseline, in 186 patients at the 2 years visit and in 169 patients at the 5 years visit.

Figure 5. Histogram of log UACR; n = 200



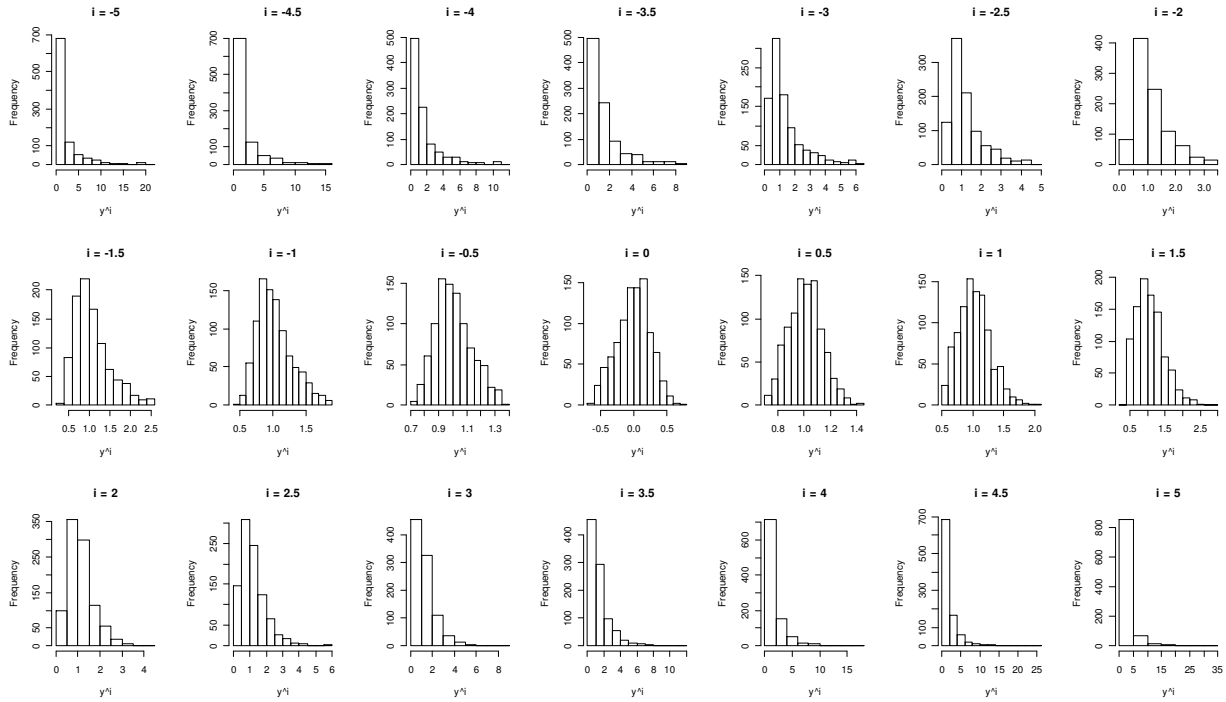
In a first step we generated 1000 normally distributed log urine albumin excretion values with a mean of 0.315 and a standard deviation of 1.766. Mean and standard deviation were taken from the sub sample of the urine albumin excretion values as mentioned above. The seed was set at 4958034. Figure 6 illustrates this first step.

Figure 6. Histogram of 1000 normally distributed values with mean 0.315 and SD 1.766



The next step was to find a suitable transformation that maps normally distributed values to a plausible distribution of urine albumin excretion values. Therefore this normally distributed values were taken to the powers from -5 to 5 at intervals of 0.5. Histograms of y^i, where i ranges from -5 to 5 and y^0 was redefined as $log(y)$ are shown in Figure 7. We see that none of these transformations really resembles the original distribution of the log urine albumin excretion values from Figure 5. Hence, there was a need for a more elaborated and targeted approach to transform UACR, the Box-Cox transformation seemed to be a suitable candidate.

Figure 7. Transformations: normally distributed values were taken to the powers ranging from -5 to 5 in 0.5 steps



In order to obtain optimal Box-Cox parameters for the simulation the `objective` function (see R functions in the appendix) and for the analysis of the ONTARGET-SysKid dataset the `target` function (see R functions in the appendix) were written. These two functions do basically the same thing: They calculate the standard deviation of the difference of the quantiles of a standard normal distribution and the standardized residuals of a random coefficients model which uses the Box-Cox transformed log urine albumin excretion values as a dependent variable. The only difference between these two functions is that they use different Box-Cox transformations (with one and two parameters, respectively). The `objective` function was minimized using R's built-in function `optim` with starting values 1 and 2 for λ and c respectively on the ONTARGET-SysKid dataset (5095 patients). Extract of the R Code for the function call:

```
optim (startvalues, objective, ret=F, x, time, ID)
# WHERE:
# startvalues ... vector of the two startvalues for lambda and c
# objective ..... R function which is minimized
# ret=F ......... only the minimal standard deviation is returned
# x ............. log UACR
# time .......... UACR measurement time points in years
# ID ............ patient's identification
```

The best Box-Cox transformation for the simulation was found at $\lambda = 0.5011719$ and $c = 2.8494141$ with a minimal standard deviation(error) of 0.04727681.[2]The results from the optimization function were used as constants in the final Box-Cox transformation function (see `boxcox` function in the appendix). In the sequel, these Box-Cox parameters will be referred to as the 'global Box-Cox parameters'.

The histogram in Figure 8 visualizes 2000 simulated and inverse-Box-Cox (using the R-function `invboxcox`, see R functions in the appendix) transformed baseline urine albumin excretion values, obtained from the `simdata` function (see subsection 4.2.1.1 and self-written R-function in the appendix). The seed was set to 3040 and all fixed effects were set to zero. Table 5 below shows the 5 %, 10 %, 25 %, 50 %, 75 %, 90 % and 95 % percentiles of this simulated urine albumin excretion values. For example, 1.23 mg/mmol is the median UACR at baseline of this example-simulation.

[2]These Box-Cox parameters were evaluated at an initial stage of this research project, where it was assumed that UACR in the ONTARGET study was expressed in mg/g. However, only later (after the main setup of the simulation study was already fixed) it turned out that the unit of measurement was actually mg/mmol. However, the correct Box-Cox parameters would have been very similar at λ = 0.4999641 and c = 2.4922054 with a minimal standard deviation(error) of 0.05077075. Therefore, we decided not to repeat simulations with the correct global Box-Cox parameters.

Figure 8. Histogram of 2000 simulated baseline UACR

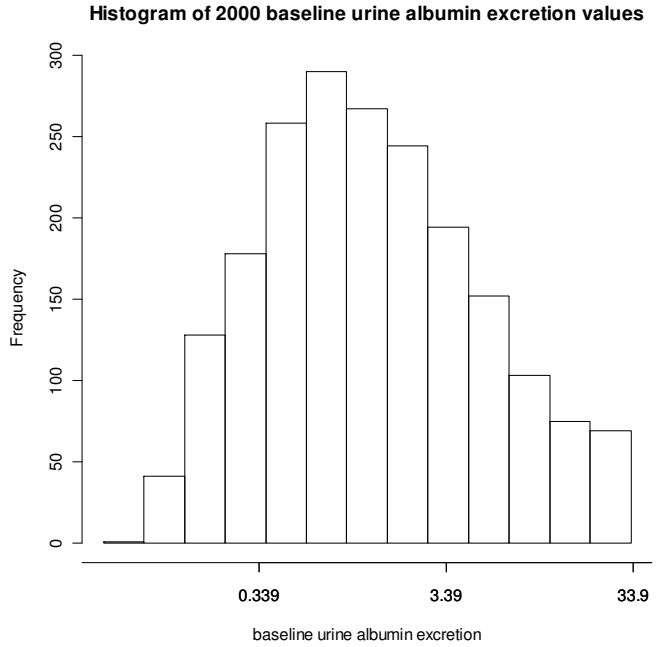


Table 5. Percentiles of 2000 simulated baseline UACR (in mg/mmol)

Percentile	5%	10%	25%	50%	75%	90%	95%
UACR at baseline	0.178	0.240	0.494	1.230	3.425	9.339	16.152

4.2 Simulation

This subsection is split into two parts: an introduction to and preparations for the simulation and the results from the simulation.

4.2.1 Preparations for simulating urine albumin excretion trajectories

Defining UACR simulation parameters, visualizing the true probability of progression, finding the optimal sample size for the simulation and an overview of the simulation process are the main topics of this subsection.

4.2.1.1 Preparations for data generation

One of the first steps before the simulation was to define the variance-covariance matrix G, which was used in the data generation process in the simulation. As a mixed model with Box-Cox transformed log albumin excretion values (BC log UACR) as outcome variable, time as fixed effect and the interaction of time with subject identification as random effect, and a similar mixed model additionally including the covariables age, BMI, sex and statin use had a very similar variance-covariance matrix G, the variance-covariance matrix and the residual sum of squares of the simpler model was used in the simulation process. Again, the result was obtained by fitting the first mentioned mixed

model on the ONTARGET-SysKid data (5095 patients). The model was obtained by executing the following R Code. The function `lme` requires the R package nlme.

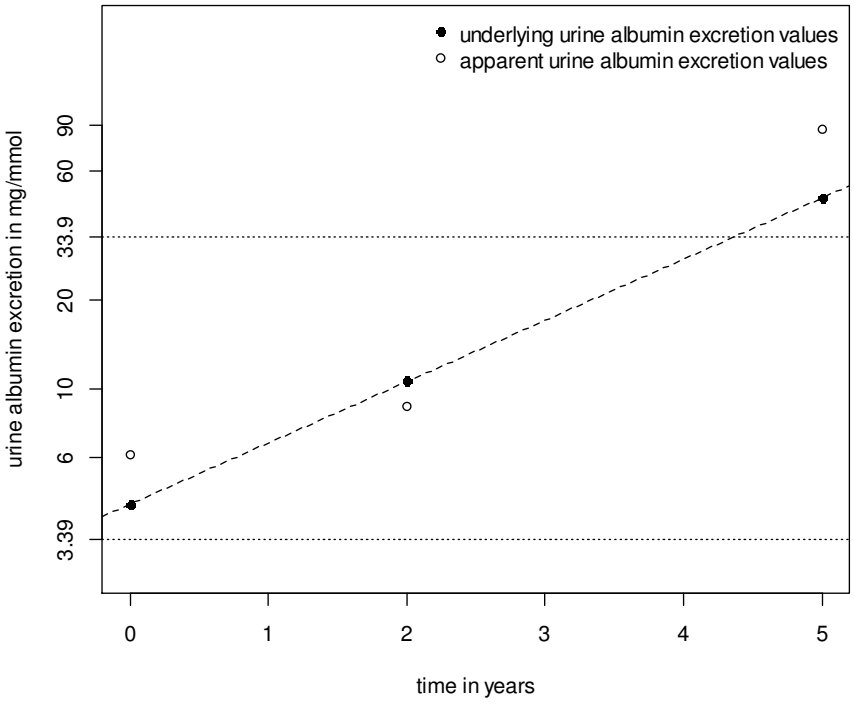
```
lme(albBC~time, random=~1+time|ID, data=ONTARGETSysKid)
# WHERE:
# the function lme is in the R package nlme
# albBC ... BC log UACR
# time .... UACR measurement time points in years
# ID ...... patient's identification
```

The residual sum of squares (RSQE) of the random coefficients model is 0.6362203 and the correlation between b_0 and b_1 is 0.0510822. The estimated covariance matrix of the random effects is given by:

$$G = \begin{pmatrix} Var(b_{0i}) & Cov(b_{0i}, b_{1i}) \\ Cov(b_{0i}, b_{1i}) & Var(b_{1i}) \end{pmatrix} = \begin{pmatrix} 0.5710800 & 0.0034829 \\ 0.0034829 & 0.0081404 \end{pmatrix}$$

This information was used in the data generation process during the simulation. In the following R-Code, which creates Figure 9, we see how a trajectory of UACR for a fictive patient with $x = 1$ can be produced by using this information.

Figure 9. Trajectory of UACR of a fictive patient of group x = 1



The seed was set at 9228, γ_{00}, γ_{10} and γ_{01} were all set to zero and γ_{11} was fixed at 0.05. The fixed intercept and slope of the BC log UACR measurements were drawn from a multivariate normal distribution with mean vector $[0, 0.05]'$ and variance-covariance matrix G. For each UACR measurement a random number, representing the residual, was drawn from a normal distribution with mean

0 and standard deviation RSQE. Putting these ingredients together leads to the simulation of the BC log UACR measurements of this fictive patient, which were used in the BCLOG models. For the log UACR, which was used in the LOG models, the simulated BC log UACR measurements were transformed using the `invboxcox` function (see appendix), which is the inverse of the Box-Cox transformation. The trajectory was plotted on the BC log UACR scale, ranging from 0 to 2.5, but using the underlying UACR in mg/mmol to label the y-axis to provide a clear picture of a patient's trajectory. Note that the tick marks on the y-axis are not regular, because the plotted BC log UACR are given on the UACR scale. Moreover, the two cut-off points are marked by two horizontal dotted lines.

```
# fixed parameters of the fictive patient:
x<-1
gamma00<-gamma01<-gamma10<-0
gamma11<-0.05
sigma0<-sigma1<-rbind(c(0.57108,0.0034829),c(0.0034829,0.0081404))
RSQE=0.6362203

# initializing variables:
# nbase, nalb2 and nalb5 are UACR measurements at baseline, 2 years and 5 years
# progr is the variable for progression: 1=yes, 0=no
coeff<-numeric(2) # fixed coefficients
time<-c(0,2,5)
library(mnormt) # required package for the function rmnorm
set.seed(9228)# seed

# draw intercept and slope:
coeff<-rmnorm(n = 1, mean = c(gamma10, gamma11), varcov=sigma1)

# simulation of BC log UACR:
nbase<- coeff[1]+rnorm(n=1, mean=0,sd=RSQE)
nalb2<- coeff[1]+ coeff[2]*2+ rnorm(n=1, mean=0,sd=RSQE)
nalb5<- coeff[1]+ coeff[2]*5+ rnorm(n=1, mean=0,sd=RSQE)

# check for progression & load R functions: invboxcox, cutpoint and progression
progr<-progression(invboxcox(x=nbase),invboxcox(x=nalb5))

# plot UACR measurements:
plot(time,c(nbase,nalb2,nalb5), ylab="urine albumin excretion in mg/mmol",
 yaxt="n", ylim=c(0,2.5), xlab="time in years",
 main="Trajectory of urine albumin excretion values of a fictive patient")
abline(coeff,lty=2) # linear trend
uacr_hat<-c(coeff[1],coeff[1]+coeff[2]*2,coeff[1]+coeff[2]*5)
points(time, uacr_hat,pch=16)
axis(side=2,at=c(nbase,nalb2,nalb5, uacr_hat,boxcox(log(c(3.39,33.9)))),
 labels=c(round(exp(invboxcox(c(nbase,nalb2,nalb5)))),
 round(exp(invboxcox(uacr_hat))),3.39,33.9))
```

```
abline(h=boxcox(log(3.39)),lty=3) # cut-off point 1
abline(h=boxcox(log(33.9)),lty=3) # cut-off point 2
legend("topright", c("underlying urine albumin excretion values",
 "apparent urine albumin excretion values"), pch=c(16,1),box.col=0)
```

Figure 9 is a typical example for a fictive patient who experiences an albuminuria class change from micro- to macroalbuminuria. The self-written R-function `progression` (for details see R functions in the appendix) evaluates whether the patient experiences during the first 5 years after randomization an event (i.e. incidence or progression). In this case, the variable `progr`, which will enter later the logistic model as outcome variable is set to 1, i.e. 'event'.

4.2.1.2 True probability of progression

Computation of the true probability of progression is outlined in subsection 3.2 and the corresponding R function `prob.progr.true` is explained in the appendix. The relationship of the true probability of progression at time point 5 regarding three different values for γ_{11} is shown in Figure 10. The true probability of progression increases slightly with γ_{11}. The impact of the baseline UACR on the probability of progression, however, is much more important. Shortly before the first cut-off point the probability of progression reaches its peak, then it decreases slightly because of the 30 % rule (see subsection 2.3). After the first cut-off point it drops almost to zero and it increases again up to the second cut-off point.

Figure 10.True probability of progression for various UACR at baseline for different γ_{11}

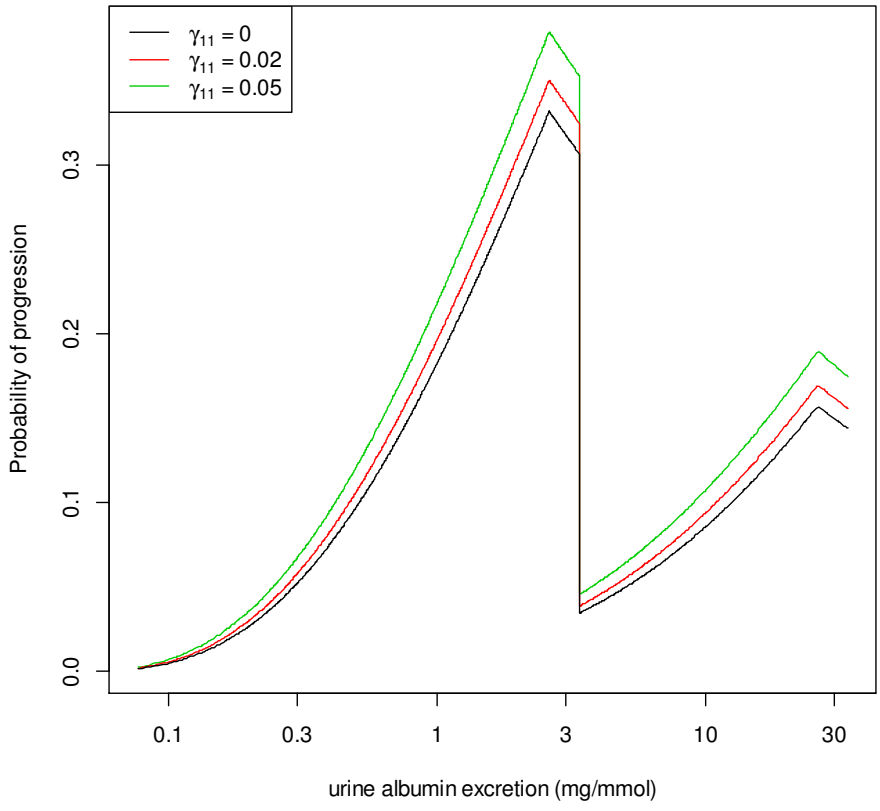


Figure 10 was created as follows: First the seed was set at 3940 and a test sample of 2000 log urine albumin excretion values was generated using the `simdata` function (see R functions in the appendix). For this example the function call of the self-written R function is:

```
testsample<-simdata(n=2000, gamma11=0.05, meanx=1)
# WHERE:
# n ........ number of patients
# gamma11... γ11
# meanx .... expected value of binary risk factor
```

Here we consider only patients from risk group $x = 1$, which is specified by `meanx = 1`. After that, the true probability of progression was calculated using the self-written function `prob.progr.true` (see R functions in the appendix).

In a next step, the mean of the true probability of progression at 5 years for these 2000 fictive patients (as approximation to the integral over the distribution of baseline values) was calculated for different values of γ_{11} and shown in Table 6.

Table 6.Mean true probability of progression for various values of γ_{11}

γ_{11}	Mean true probability of progression at 5 years
0.00	0.142
0.01	0.147
0.02	0.152
0.03	0.158
0.04	0.163
0.05	0.169
0.06	0.175
0.07	0.181
0.08	0.187
0.09	0.193
0.10	0.200
0.20	0.270

Note that with an increasing γ_{11}, which is the slope of group $x = 1$ per year, also the mean true probability of progression increases.

4.2.1.3 Searching the optimal sample size for simulation

First of all, an 'optimal' sample size was defined as a sample size where with an expected relative risk of approximately 1.2, which is the case for $\gamma_{11} = 0.05$, the power to reject $\alpha_1 = 0$ in the logistic model lies with 95% probability within the interval [0.775, 0.825].

In order to find a suitable sample size fulfilling this definition, we simulated datasets with different sample sizes, ranging from 600 to 1200 with a step size of 50, ten datasets each, and evaluated at

each sample size the power (the proportion of significant tests of $\alpha_1 = 0$) of the logistic model. A new sample size n^* was then calculated as the predicted sample size at which a power of 80% is expected. This sample size, n^*, was estimated by estimating a logistic regression model with 'significance' as the binary outcome variable and sample size as the only continuous predictor. Ten new datasets of size n^* were simulated, and the results added to the previous results. Then, n^* was recalculated, iterating this process fifty times, or stopping earlier if the 95% prediction interval for the power at n^* was entirely contained in [0.775, 0.825]. This approach is implemented in the R Code see 20120427_determinesaplesize_forbatch.r, to be found in the subsection 'supplementary files' in the appendix.

We repeated the above mentioned method five times with different seeds. Figures 11 to 15 show the convergence of sample size and upper and lower confidence limits for the power of the logistic model for each randomly chosen seed during approximately 50 iterations. Results for the optimal sample size were in the range [1023, 1067].

Figure 11.Red: lower and upper bound of the power (in %); black: sample size; seed: 4920; optimal n: 1042

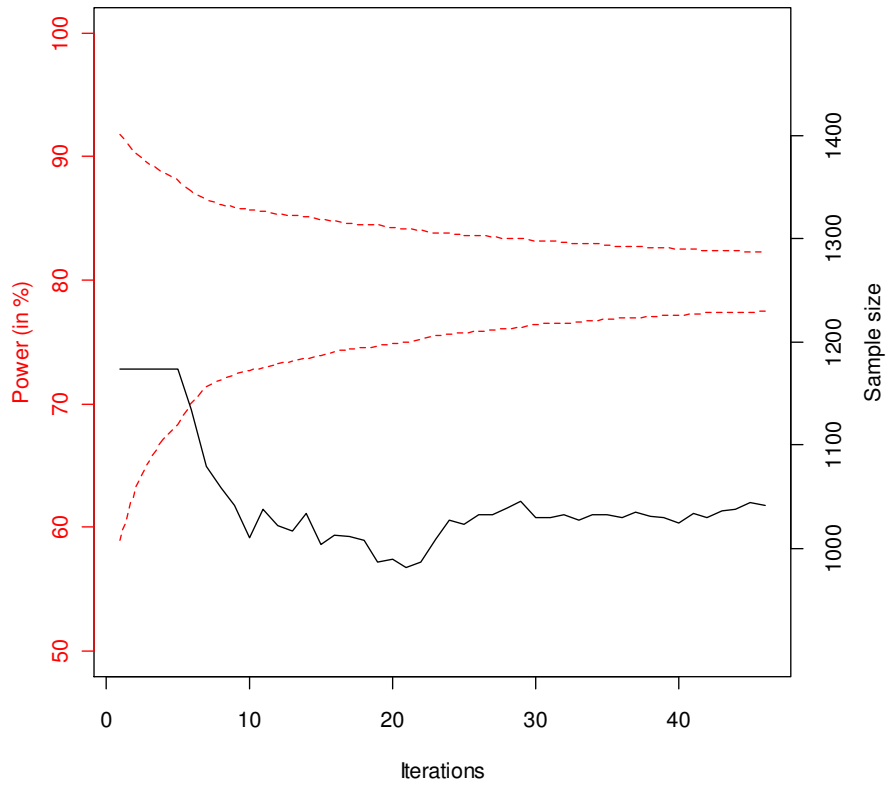


Figure 12. Red: lower and upper bound of the power (in %); black: sample size; seed: 245, optimal n: 1046

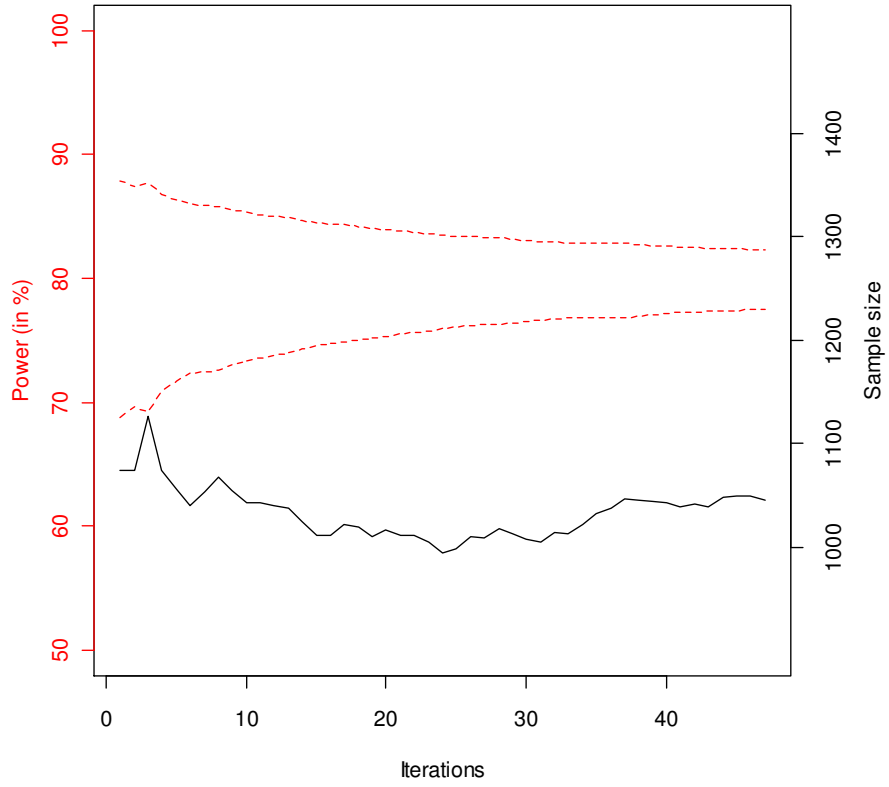


Figure 13. Red: lower and upper bound of the power (in %); black: sample size; seed: 693; optimal n: 1028

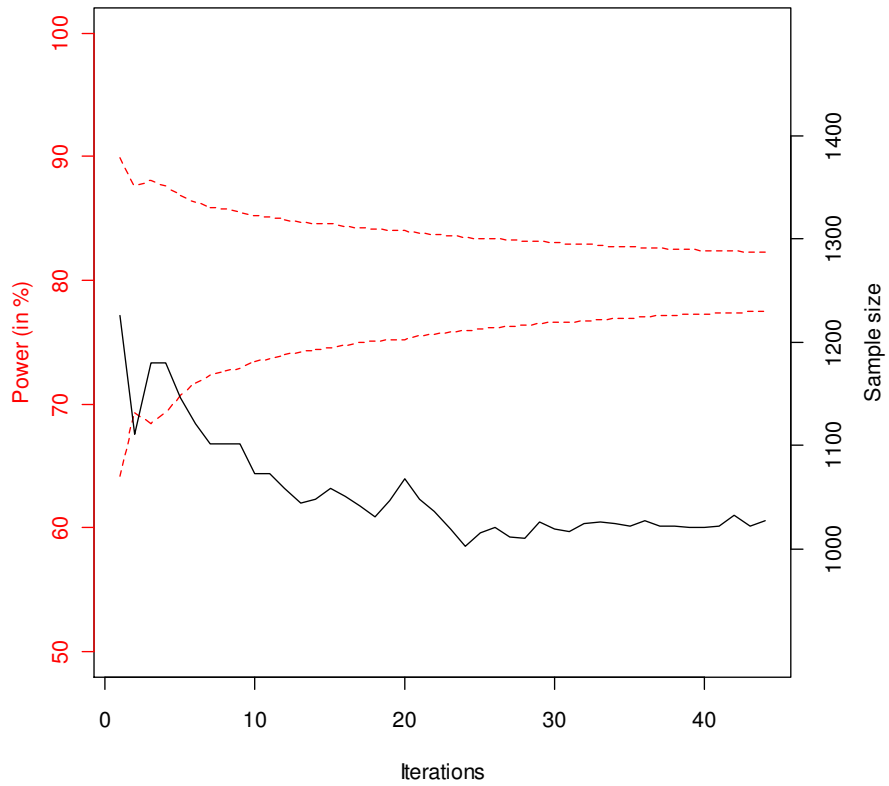


Figure 14.Red: lower and upper bound of the power (in %); black: sample size; seed: 1239; optimal n: 1067

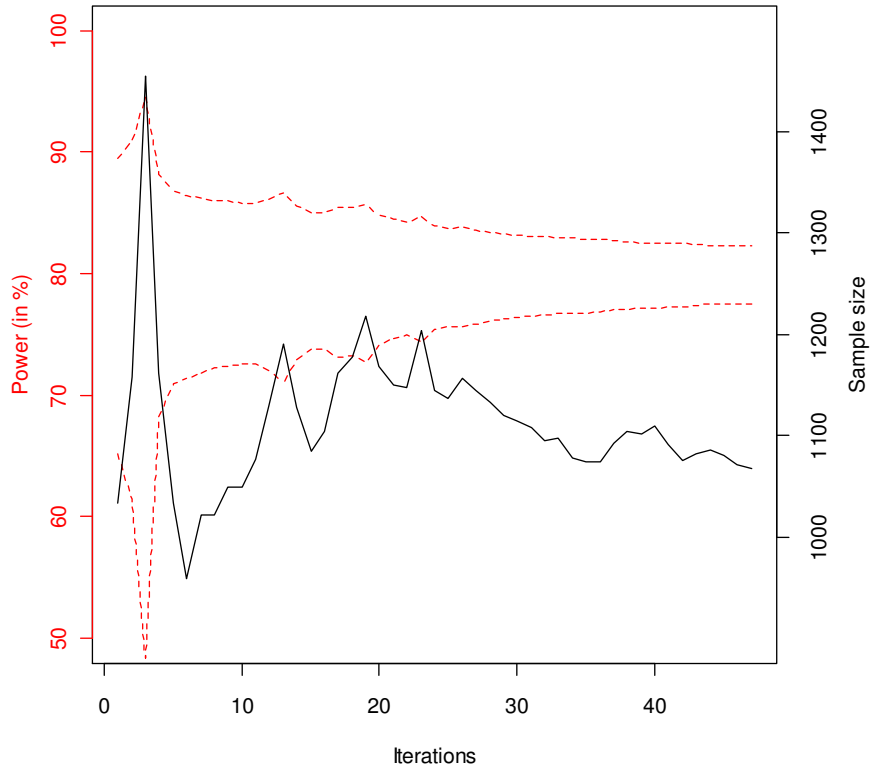
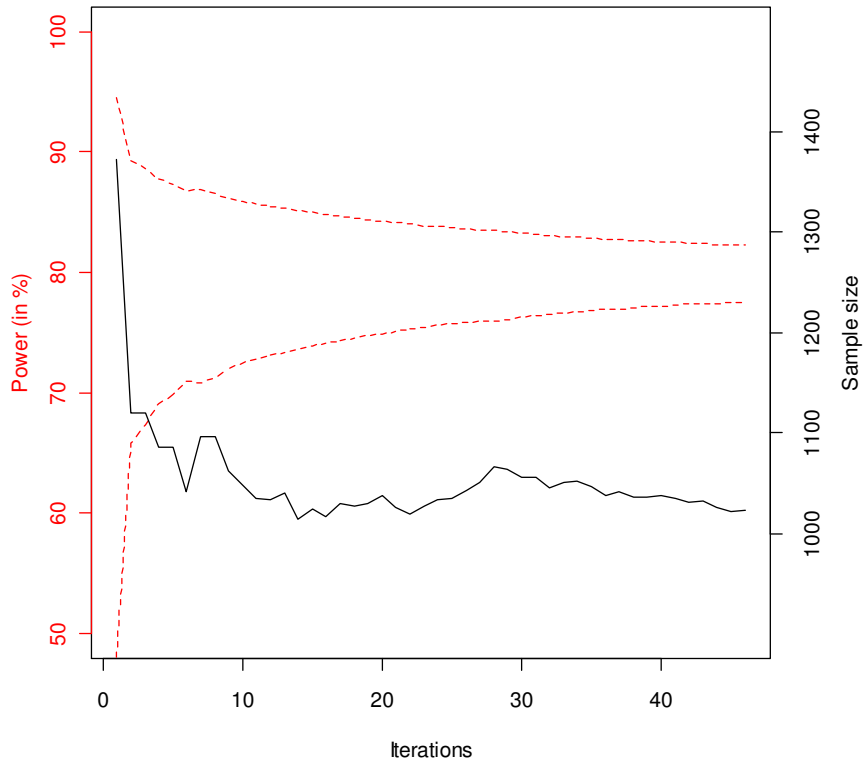


Figure 15.Red: lower and upper bound of the power (in %); black: sample size; seed: 2947; optimal n: 1023



The optimal sample size was chosen to be 1041, which is the mean of the five optimal sample sizes, which were found above (Figures 11 to 15).

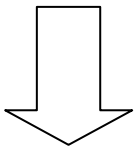
4.2.1.4 Simulation overview

Figure 16 provides an overview of the simulation process including the employed R functions.

Figure 16. Simulation overview

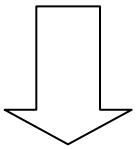
Data generation

Step	Task	R functions
1.	Simulate patients with trajectories of 3 UACR measurements (training sample)	`simdata`
2.	Transform the simulated Box-Cox log UACR	`invboxcox`



Data analysis

Step	Task	R functions
3.	Find optimal Box-Cox parameters for each simulated dataset	`boxcox2, objective, optim`
4.	Transform the log UACR using the local Box-Cox parameters	`boxcox2`
5.	Fit the logistic and the longitudinal models	`glm, lme`
6.	Calculate the progression rates of both groups of x, the apparent relative risk and the power of the risk factor x for progression in the logistic and the longitudinal models on each dataset	`progression, cutpoint`
7.	Calculate the probability of progression estimated by the logistic and longitudinal models for each dataset	`predict, progr.lme`
8.	Calculate the true probability of progression for both groups of x	`prob.progr.true`



Assessment of model improvement

Step	Task	R functions
9.	Simulate new patients (test sample)	`simdata`
10.	Calculate c-statistics and IDI by applying the logistic and longitudinal models estimated in Step 5 to the new dataset and comparing with the true progression status.	`compute.c, compute.idi`

Steps 3 and 4 of the simulation were only executed in two scenarios because of computational reasons (see subsection 4.2.2.2). For all other scenarios the global Box-Cox parameters (see subsection 4.1) were used to transform the log UACR. All mentioned R functions are described in the appendix.

4.2.2 Results

The simulation results, in which the above mentioned global Box-Cox parameters were used in the BCLOG models, is given in subsection 4.2.2.1. The results, in which for each simulated dataset the optimal Box-Cox parameters were evaluated and used in the BCLOG models, are shown in subsection 4.2.2.2. Figures for the probability of progression for each and every scenario are to be found in the appendix.

All simulated datasets were evaluated with regard to their apparent progression rates, the apparent relative risk, the power of a test of the null hypothesis $\gamma_{11} = 0$, i.e., of testing an association of the risk factor x with progression, the progression probabilities estimated by the logistic and longitudinal models for group $x = 0$ vs. group $x = 1$, the true probability of progression, expected relative risk and expected odds ratio. Apparent progression rates are the percentage of patients with progression from group $x = 1$ (analogously fro group $x = 0$). The apparent relative risk is defined as the mean over 1000 simulations of the ratio of apparent progression rates of group $x = 1$ and group $x = 0$. The power to detected an effect of the risk factor x is defined as the percentage of the significance of x in a model over 1000 simulations at a significance level of 5 %. The progression probabilities estimated by the logistic model, by the longitudinal model and the true progression probabilities are calculated as the mean over all patients of the corresponding probabilities. The expected relative risk is defined as the mean over 1000 simulations from the mean of the relative risks obtained for each patient by taking the ratio of his/her true progression probabilities that result from inserting $x = 0$ or group $x = 1$ into the true data generating model equation. The expected odds ratio is given as the mean over 1000 simulations of the mean over all patients of the individual odds ratios, where the individual odds ratios are defined as the odds of progression for group $x = 1$ divided by the odds of progression for group $x = 0$. The odds of progression for group $x = 1$ are the ratio of the true progression probability and one minus this probability for group $x = 1$ (analogously for group $x = 0$).
BCLOG model is the benchmark model as its specification matches exactly the data generating model.

4.2.2.1 No optimization of BC parameters

First of all, some general information of the simulated datasets (i.e. apparent progression rates and apparent relative risk) is given (see Table 7). Then, the logistic and longitudinal models are compared with regard to their powers and estimated probabilities of progression in a multitude of Tables and Figures. The estimated coefficients of the longitudinal models are given for two scenarios as well as their transformation from interaction parametrization to groupwise parametrization is shown in subsection 4.2.2.1.1. And finally, the model improvement of the logistic and longitudinal models is assessed by c-statistics and IDI (see subsection 4.2.2.1.2).

For simplicity, we assumed that all coefficients were zero (γ_{00} = 0, γ_{01} = 0, γ_{10} = 0) except for γ_{11}, the slope of group $x = 1$, which was varied from 0 to 0.05 by 0.01. For each value of γ_{11} 1000 datasets with 1041 fictive patients were simulated. Additionally, we looked at two different sample sizes: For $\gamma_{11} = 0.02$ the sample was as well doubled ($n = 2082$) and for $\gamma_{11} = 0.05$ the sample was also halved ($n = 520$). The seed was always set to 3294.

Table 7 shows the apparent progression rates and the resulting apparent relative risk for various values of γ_{11}. The apparent progression rate of group $x = 0$ is constant, because always the same seed was used when running the simulation function for different γ_{11}. The only exceptions are that for $n = 2082$ and $\gamma_{11} = 0.02$ we find a smaller standard deviation and for $n = 520$ and $\gamma_{11} = 0.05$ we observe a higher standard deviation of the apparent progression rate. In group $x = 1$ the apparent progression rates increase with γ_{11}. We further see that as γ_{11} increases, also the apparent relative risk increases. Only, when the sample size is doubled ($n = 1041$ and $n = 2082$, respectively) then apparent relative risk decreases from 1.145 to 1.130.

Table 7. Progression rates for various values of γ_{11} and sample sizes (n)

n	γ_{11}	Apparent progression rates*		Mean relative risk
		Group x = 0	Group x = 1	
1041	0.00	0.179 (0.017)	0.176 (0.017)	0.990
1041	0.01	0.179 (0.017)	0.189 (0.017)	1.067
1041	0.02	0.179 (0.017)	0.203 (0.018)	1.145
2082	0.02	0.180 (0.012)	0.202 (0.012)	1.130
1041	0.03	0.179 (0.017)	0.218 (0.018)	1.227
1041	0.04	0.179 (0.017)	0.233 (0.019)	1.312
520	0.05	0.180 (0.024)	0.247 (0.028)	1.397
1041	0.05	0.179 (0.017)	0.248 (0.019)	1.399

*mean (sd) over 1000 simulated datasets

Table 8 demonstrates that the power to detect an effect of x increases with higher values of γ_{11} for the logistic and the longitudinal models. However the power of the longitudinal models increases faster and is always higher than the power of the logistic models. For γ_{11} = 0.05 the power of the risk factor x for progression in the longitudinal model which uses Box-Cox-transformed log urine albumin excretion values (BCLOG model), reaches almost 100 %, while the power for the logistic model for the same value of γ_{11} is only 81 %. When we compare the simulation results in the case of $\gamma_{11} = 0.05$ for the sample sizes $n = 1041$ and $n = 520$, we observe that a reduction in sample size leads to a strong reduction of the power of the risk factor for progression estimated by both models, the logistic and the longitudinal. The power of the logistic model was reduced from 81 % to only 50 % and the power of the longitudinal model dropped from 99 % to 83 %. Comparing the results in the case of $\gamma_{11} = 0.02$ for the sample sizes $n = 1041$ and $n = 2082$ shows that doubling the sample size leads to an augmentation of the power of the risk factor for progression estimated by both models, the logistic and the longitudinal. The power of the logistic model was raised from 17 % to only 26 % and the power of the longitudinal model lifted up from 39 % to 70 %.

Table 8. Power to detect an effect of x for various values of γ_{11} and sample sizes (n)

n	γ_{11}	Power*			
		Logistic		Longitudinal	
		BCLOG model	LOG model	BCLOG model	LOG model
1041	0.00	0.056	0.056	0.055	0.062
1041	0.01	0.069	0.068	0.133	0.113
1041	0.02	0.171	0.176	0.386	0.342
2082	0.02	0.262	0.267	0.699	0.589
1041	0.03	0.357	0.359	0.708	0.658
1041	0.04	0.605	0.613	0.909	0.885
520	0.05	0.498	0.496	0.828	0.797
1041	0.05	0.806	0.811	0.985	0.974

* percentage of the significance of x in corresponding model over 1000 simulations

Tables 9 and 10 show that the estimated probability of progression increases with increasing γ_{11} in the logistic as well as in the longitudinal model and that these probabilities are always higher in the logistic models than in the longitudinal models. For example, for a fictive patient of group $x = 1$ with a simulated slope of 0.05, the probability of progression is estimated to be 24.7 % in a logistic model and only 16 % in a longitudinal model. And for a fictive patient of group $x = 0$ with a simulated slope of 0.02, the probability of progression estimated by the logistic model is 17.9 %, whereas it is only 13.6 % when estimated by the longitudinal model. Moreover, the probability of progression in group $x = 1$ is always higher or equal than in group $x = 0$ for all models.

Furthermore, we see in Tables 9 and 10 that there is hardly a difference between the estimation of the probability of progression through Box-Cox log transformed UACR (BCLOG model) or log transformed UACR (LOG model). The variation is slightly higher in the misspecified LOG models.

Another interesting thing, which one might not see at the first sight, is, that the probability of progression estimated by the longitudinal LOG model for patients of risk group $x = 0$ increases as γ_{11} increases. This should not be the case due to two reasons: First, since our model formula is $Y_{ti} = \beta_{11} \cdot x \cdot t + b_{0i} + b_{1i} \cdot t + \varepsilon_{ti}$ and $x = 0$, there should be no influence of γ_{11} and second, the same seed for the simulation of each γ_{11} was used. Furthermore, we notice that this is only the case for the longitudinal LOG model. The logistic BCLOG and LOG models as well as the longitudinal BCLOG model do not show an increase in the probability of progression for risk group $x = 0$ as γ_{11} increases. The explanation is, that the LOG model is misspecified, because the residuals of this model are not normally distributed. Therefore, one important model assumption is violated and this results in the bias, which we see in Table 10.

Tables 9 and 10 further show that reducing the sample size by half for $\gamma_{11} = 0.05$, from 1041 to 520, or doubling it for $\gamma_{11} = 0.02$, from 1041 to 2082, has no effect on the probability of progression estimated by neither the logistic model nor the longitudinal model.

Table 9. Logistic models: Probability of progression considering various values of γ_{11}

n	γ_{11}	Estimated probability of progression in the population†			
		Logistic model			
		$x = 0$		$x = 1$	
		BCLOG model	LOG model	BCLOG model	LOG model
1041	0.00	0.179 (0.168, 0.190)	0.178 (0.168, 0.190)	0.175 (0.164, 0.186)	0.175 (0.164, 0.186)
1041	0.01	0.179 (0.168, 0.190)	0.179 (0.168, 0.190)	0.189 (0.177, 0.200)	0.189 (0.177, 0.200)
1041	0.02	0.179 (0.168, 0.190)	0.179 (0.168,0.190)	0.203 (0.191, 0.214)	0.203 (0.191,0.214)
2082	0.02	0.179 (0.168, 0.190)	0.179 (0.168, 0.190)	0.203 (0.191, 0.214)	0.203 (0.191, 0.214)
1041	0.03	0.179 (0.168, 0.190)	0.178 (0.168,0.190)	0.217 (0.206, 0.229)	0.217 (0.205,0.229)
1041	0.04	0.179 (0.168, 0.190)	0.178 (0.168,0.190)	0.232 (0.221, 0.244)	0.232 (0.220,0.244)
520	0.05	0.179 (0.168, 0.190)	0.178 (0.168, 0.190)	0.247 (0.235, 0.260)	0.247 (0.235, 0.260)
1041	0.05	0.179 (0.168, 0.190)	0.178 (0.168,0.190)	0.247 (0.235, 0.260)	0.247 (0.235,0.260)

† median (1st and 3rd Quartile) over 1000 simulated datasets

Table 10. Longitudinal models: Probability of progression considering various values of γ_{11}

n	γ_{11}	Estimated probability of progression in the population†			
		Longitudinal model			
		$x = 0$		$x = 1$	
		BCLOG model	LOG model	BCLOG model	LOG model
1041	0.00	0.136 (0.130,0.142)	0.134 (0.127, 0.142)	0.136 (0.129,0.142)	0.134 (0.127,0.141)
1041	0.01	0.136 (0.130,0.142)	0.135 (0.128, 0.143)	0.141 (0.134,0.146)	0.139 (0.132, 0.146)
1041	0.02	0.136 (0.130,0.142)	0.136 (0.129, 0.144)	0.145 (0.138,0.151)	0.144 (0.137, 0.152)
2082	0.02	0.136 (0.130, 0.142)	0.136 (0.129, 0.144)	0.145 (0.138, 0.151)	0.144 (0.137, 0.152)
1041	0.03	0.136 (0.130,0.142)	0.137 (0.130, 0.145)	0.150 (0.143,0.156)	0.149 (0.142, 0.157)
1041	0.04	0.136 (0.130,0.142)	0.138 (0.130, 0.146)	0.155 (0.148,0.161)	0.155 (0.147, 0.163)
520	0.05	0.136 (0.130, 0.142)	0.139 (0.131, 0.146)	0.160 (0.152, 0.166)	0.160 (0.152, 0.168)
1041	0.05	0.136 (0.130,0.142)	0.139 (0.131, 0.146)	0.160 (0.152,0.166)	0.160 (0.152, 0.168)

† median (1st and 3rd Quartile) over 1000 simulated datasets

The true probability of progression, the expected relative risk and the expected odds ratio of progression are described in Table 11. The expected relative risk for progression for group $x = 1$ increases with γ_{11}. Since the expected odds ratio is always higher than one when increasing γ_{11}, we notice that the true probability of progression is higher in group $x = 1$ than in group $x = 0$. For example, for a fictive patient with a simulated slope of 0.05, the expected odds ratio of progression is 1.279 times higher in group $x = 1$ than group $x = 0$.

Table 11. True probability of progression considering various values of γ_{11}

n	γ_{11}	True probability of progression†		Expected RR	Expected OR
		$x = 0$	$x = 1$		
1041	0.00	0.139 (0.137, 0.141)	0.139 (0.137, 0.141)	1.000	1.000
1041	0.01	0.139 (0.137, 0.141)	0.144 (0.142,0.146)	1.044	1.051
1041	0.02	0.139 (0.137, 0.141)	0.149 (0.147,0.151)	1.090	1.104
1041	0.03	0.139 (0.137, 0.141)	0.154 (0.152,0.157)	1.137	1.160
1041	0.04	0.139 (0.137, 0.141)	0.160 (0.158,0.162)	1.186	1.218
1041	0.05	0.139 (0.137, 0.141)	0.166 (0.163,0.168)	1.237	1.279

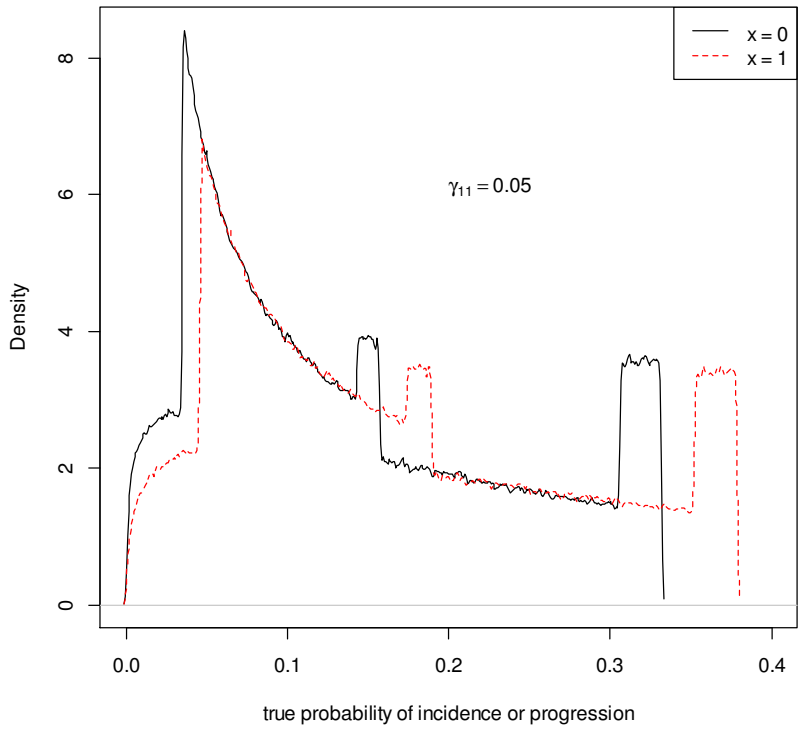
† median (1st and 3rd Quartile) over 1000 simulated datasets

Tables 9 to 11 reveal that the probability of progression estimated by the longitudinal model comes closer to the true probability of progression than the one estimated by the logistic model.

Figure 17 shows the density of the true probability of progression over all fictive patients who were simulated with a γ_{11} of 0.05. Since in each of the 1000 simulation runs 1041 fictive patients were simulated, the total number of patients in this Figure is a bit more than one million. However, the density of the true probability of progression is asymmetric and has the same shape for both groups

of x, i.e. both have three peaks, which is probably due to the two cut-off points of UACR and the 30 % rule. The main difference for patients with (red broken line) and without (black line) the risk factor x is that the true probability of progression is higher for exposed subjects. The bandwidth was set to 0.0004.

Figure 17. Density of true probability of progression over all simulated patients (n = 1041000); γ_{11} = 0.05



4.2.2.1.1 Coefficients of the longitudinal models

Tables 12 to 16 give an overview of the two different parameterization of the parameters of the longitudinal models from the simulation ($n = 1041$). In Tables 12 and 14 for $\gamma_{11} = 0$ and in Tables 15 and 16 $\gamma_{11} = 0.05$. How the interaction parametrization can be transformed into the groupwise parametrization (γ_{00}, γ_{01}, γ_{10} and γ_{11}) is shown in Table 13.

Table 12. Estimated coefficients by interaction parameterization; $\gamma_{11} = 0$

	$\widehat{\beta}_0$*	$\widehat{\beta}_1$*	$\widehat{\beta}_2$*	$\widehat{\beta}_3$*
BCLOG model	-0.020 (0.038)	0.004 (0.009)	0.001 (0.054)	0.000 (0.012)
LOG model	0.326 (0.057)	0.021 (0.013)	0.004 (0.081)	-0.002 (0.012)

* values are given by mean (sd) over 1000 simulated datasets

Table 13. Groupwise parametrization vs. interaction parametrization

Groupwise p.	γ_{00}	γ_{01}	γ_{10}	γ_{11}
Interaction p.	β_0	β_2	$\beta_0+\beta_1$	$\beta_2+\beta_3$

Applying the notation change from Table 13 to Table 12 leads to the estimated coefficients of the longitudinal models in the groupwise parametrization (see Table 14).

Table 14. Estimated coefficients by groupwise parameterization; $\gamma_{11} = 0$

	$\widehat{\gamma}_{00}$*	$\widehat{\gamma}_{01}$*	$\widehat{\gamma}_{10}$*	$\widehat{\gamma}_{11}$*
BCLOG model	-0.020 (0.038)	0.001 (0.054)	-0.015 (0.037)	0.001 (0.052)
LOG model	0.326 (0.057)	0.004 (0.081)	0.347 (0.055)	0.002 (0.078)

* values are given by mean (sd) over 1000 simulated datasets

As it is shown in Table 12 and 15, the estimated coefficients of longitudinal models in which γ_{11} was set to 0.05 in the simulation are very similar to those where γ_{11} was set to 0. As expected only β_3 changes in the BCLOG model. In the LOG model β_2 also changes slightly.

Table 15. Estimated coefficients by the interaction parametrization; $\gamma_{11} = 0.05$

	$\widehat{\beta}_0$*	$\widehat{\beta}_1$*	$\widehat{\beta}_2$*	$\widehat{\beta}_3$*
BCLOG model	-0.020 (0.038)	0.004 (0.009)	0.001 (0.054)	0.050 (0.012)
LOG model	0.326 (0.057)	0.021 (0.013)	0.003 (0.081)	0.076 (0.019)

* values are given by mean (sd) over 1000 simulated datasets

Table 16. Estimated coefficients by groupwise parameterization; $\gamma_{11} = 0.05$

	$\widehat{\gamma}_{00}$*	$\widehat{\gamma}_{01}$*	$\widehat{\gamma}_{10}$*	$\widehat{\gamma}_{11}$*
BCLOG model	-0.020 (0.038)	0.001 (0.054)	-0.015 (0.037)	0.051 (0.052)
LOG model	0.326 (0.057)	0.003 (0.081)	0.347 (0.055)	0.079 (0.078)

* values are given by mean (sd) over 1000 simulated datasets

4.2.2.1.2 Assessing model improvement

C-statistics and integrated discrimination improvement(IDI) were used for assessing model improvement. In order to calculate the c-statistics and the IDI of the logistic and longitudinal models in R, new functions were written (see R functions in the appendix). Furthermore, all models were calculated twice in the simulation, once without the marker x and once with the marker x.

4.2.2.1.2.1 C-Statistics

Table 17 and 18 compare the c-statistic and the correlation between progression and its estimation by the logistic model to the c-statistic and to the correlation between progression and its estimation by the longitudinal model. This was achieved through 1000 simulation runs for each γ_{11}, using the fit objects of the logistic and the longitudinal models, which were generated in the previous simulation. The seed was set at 3239.

Table 17 shows that the c-statistics for the longitudinal model are slightly higher than the c-statistics of the logistic model. This is the case for both models, BCLOG and LOG. Thus, in general, the longitudinal model is more accurate in estimating the probability of progression than the logistic model. Moreover, increasing γ_{11} does not really affect the c-statistics. In the longitudinal models it has no effect at all (except on the misspecified LOG models) and in the logistic models the c-statistic increases only barely. Furthermore, we observe that the mean c-statistics of the simulation, where $\gamma_{11} = 0.02$, are not affected by a increase in sample size from 1041 to 2082. Only their standard deviations decrease slightly (from approx. 0.020 to 0.014). When we express the standard deviations as variance, we see that the variance of the c-statistics is decreased by half as the sample size is doubled: 0.0004 for $n = 1041$ and 0.0002 for $n = 2082$. This is good, since the one divided by the variance is proportional to the sample size. The same is the case for a reduction in sample size from 1041 to 520. Then the standard deviation of the c-statistics increases proportionally with the sample size (from approx. 0.02 to 0.03).

Table 17. C-statistics of all models for various values of γ_{11}

n	γ_{11}	C-Statistic*			
		Logistic model		Longitudinal model	
		BCLOG model	LOG model	BCLOG model	LOG model
1041	0.00	0.675 (0.021)	0.675 (0.021)	0.695 (0.020)	0.688 (0.021)
1041	0.01	0.674 (0.021)	0.675 (0.021)	0.695 (0.020)	0.688 (0.021)
1041	0.02	0.674 (0.021)	0.675 (0.021)	0.695 (0.020)	0.688 (0.021)
2082	0.02	0.675 (0.014)	0.675 (0.014)	0.694 (0.014)	0.688 (0.014)
1041	0.03	0.675 (0.021)	0.676 (0.020)	0.694 (0.020)	0.688 (0.021)
1041	0.04	0.677 (0.020)	0.677 (0.020)	0.695 (0.020)	0.689 (0.020)
520	0.05	0.678 (0.028)	0.679 (0.028)	0.696 (0.028)	0.689 (0.028)
1041	0.05	0.679 (0.020)	0.679 (0.020)	0.695 (0.019)	0.689 (0.020)

* values are given by mean (sd) over 1000 simulated datasets

Table 18 demonstrates that the correlation of predicted probabilities of progression and observed progression status in the longitudinal models is slightly higher than it is in the logistic models. For example, the mean correlation over 1000 simulation runs is for $\gamma_{11} = 0$ in the longitudinal model approximately equal to the mean correlation over 1000 simulation runs for $\gamma_{11} = 0.05$ in the logistic model. Furthermore, as γ_{11} increases also the correlation between progression and its probability estimation increases. This is the case for both models, logistic and longitudinal. Again, reducing (or increasing) the sample size leads to no changes in the mean, but an increase (or reduction) in the standard deviation.

Table 18. Correlation of predicted probabilities of progression and observed progression status for different γ_{11}

n	γ_{11}	Correlation*†			
		Logistic model		Longitudinal model	
		BCLOG model	LOG model	BCLOG model	LOG model
1041	0.00	0.232 (0.029)	0.232 (0.030)	0.258 (0.031)	0.251 (0.032)
1041	0.01	0.234 (0.029)	0.234 (0.030)	0.261 (0.031)	0.254 (0.032)
1041	0.02	0.238 (0.029)	0.238 (0.030)	0.264 (0.031)	0.257 (0.031)
2082	0.02	0.239 (0.020)	0.239 (0.020)	0.264 (0.021)	0.258 (0.022)
1041	0.03	0.243 (0.029)	0.243 (0.030)	0.268 (0.031)	0.261 (0.032)
1041	0.04	0.250 (0.029)	0.249 (0.030)	0.272 (0.031)	0.265 (0.032)
520	0.05	0.255 (0.042)	0.256 (0.043)	0.279 (0.044)	0.272 (0.045)
1041	0.05	0.257 (0.029)	0.256 (0.030)	0.277 (0.031)	0.270 (0.031)

* values are given by mean (sd) over 1000 simulated datasets

† of predicted probabilities of progression and observed progression status

4.2.2.1.2.2 Integrated Discrimination Improvement

Table 19 compares the IDI of the logistic models to those of the longitudinal models considering various values of γ_{11} in 1000 simulation runs. With an increasing γ_{11} also the IDI of both models increases. The IDI of the logistic models increases slightly faster than the IDI of the longitudinal models and it is always higher than the IDI in the longitudinal models. Moreover, reducing (or increasing) the sample size leads to no changes in the mean, but a proportional increase (or reduction) in the standard deviation of the IDI.

Table 19. IDI of all models considering various values of γ_{11}

n	γ_{11}	IDI*†			
		Logistic model		Longitudinal model	
		BCLOG model	LOG model	BCLOG model	LOG model
1041	0.00	0.0001 (0.0012)	0.0001 (0.0011)	0.0000 (0.0003)	0.0000 (0.0004)
1041	0.01	0.0003 (0.0014)	0.0003 (0.0013)	0.0001 (0.0004)	0.0001 (0.0004)
1041	0.02	0.0011 (0.0020)	0.0010 (0.0019)	0.0004 (0.0007)	0.0004 (0.0006)
2082	0.02	0.0010 (0.0012)	0.0009 (0.0011)	0.0004 (0.0004)	0.0004 (0.0004)
1041	0.03	0.0026 (0.0028)	0.0025 (0.0026)	0.0010 (0.0009)	0.0009 (0.0008)
1041	0.04	0.0049 (0.0037)	0.0046 (0.0034)	0.0017 (0.0012)	0.0016 (0.0011)
520	0.05	0.0075 (0.0063)	0.0069 (0.0059)	0.0027 (0.0021)	0.0025 (0.0019)
1041	0.05	0.0079 (0.0046)	0.0073 (0.0043)	0.0027 (0.0015)	0.0025 (0.0013)

* values are given by mean (sd) over 1000 simulated datasets

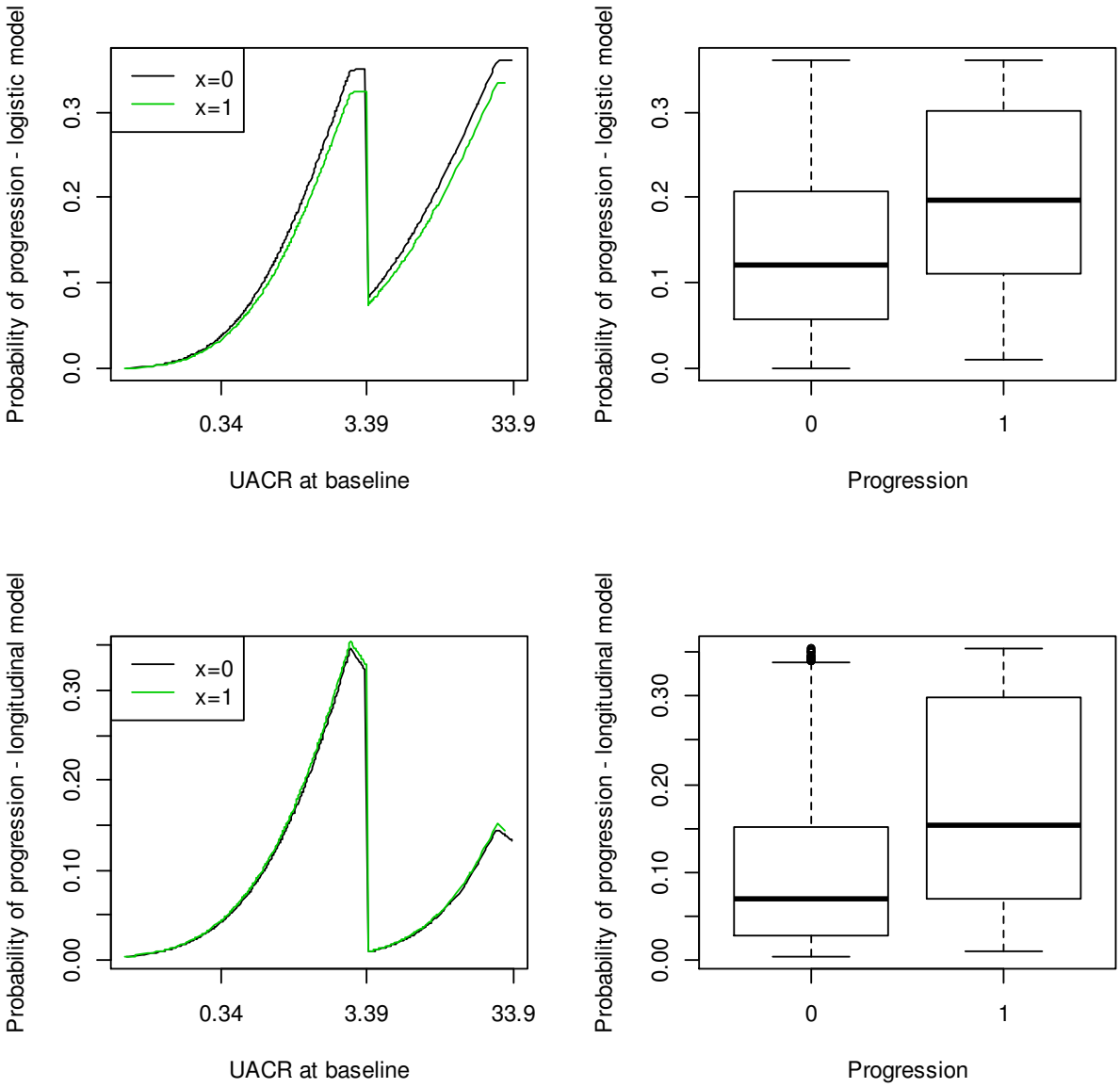
† of the logistic and longitudinal model for the marker x

The differences in the logistic and longitudinal models are all significant (p-value < 0.01) for both, the mean c-statistics and mean IDI, when a applying a two-sided t-test.

4.2.2.1.2.3 Plausibility of IDI results

In the following, the plausibility of the IDI results is investigated in the LOG models assuming the values 0 and 0.05 for γ_{11}. Figure 18 visualizes the probability of progression for the LOG models from the simulation in which γ_{11} was set to 0. In the left panels, the probability of progression is shown for both groups of x and simulated baseline UACR ranging from 0 to 33.9. In the right panels, the probability of progression is visualized in boxplots for patients without progression (progression = 0) and with a progression in CKD (progression = 1). The two plots above concern the logistic LOG model and the two plots below the longitudinal LOG model. The seed was set at 3432 and the fit objects from simulation number 346, which was chosen randomly, from the training's sample were used.

Figure 18.Probability of progression: in dependence of UARC at baseline (left panel) and progression (right panel); estimated by the logistic LOG model (top panel) and by the longitudinal LOG model (bottom panel); γ_{11}=0; seed = 3432; training sample no. = 346



As Figure 18 shows, the probability of progression estimated by the logistic model for patients belonging to group $x = 1$ is lower than for patients of group $x = 0$. By contrast, the probability estimation by the longitudinal model is nearly the same, if not even reversed, for both groups of x. Furthermore, the boxplots visualize that the estimation of the probability of progression is not perfect since the two boxes overlap to a large degree. This is the case for both models, the logistic and the longitudinal.

In the scenario of Figure 18, the IDI for the longitudinal model is zero. In the following the calculation of the IDI in R is explained in 7 steps. The parameter `fitl` is the fit object of the longitudinal model without the marker x and `fitlx` is the fit object of the longitudinal model with the marker x.

```
# Step 1: Calculate the probability of progression from the
# longitudinal LOG model without the risk factor x (old model); n=1041
# Notation: lme ... longitudinal model

for(i in 1:n)
{
  problme[i]<-progr.lme(fitl, model=modeltype,time=5,X=F,
logbase=invboxcox(x=newdata$nbase[i]))$prob.progr
}
# Step 2: Calculate the integrated sensitivity of the old model
ISold_lme<-mean(problme[newdata$progr==1],na.rm=T)

# Step 3: Calculate one minus integrated specifity of the old model
IPold_lme<-mean(problme[newdata$progr==0],na.rm=T)

# Step 4: Calculate the probability of progression from the
# longitudinal LOG model which includes the risk factor x (new model); n=1041
for(i in 1:n)
{
  problmex[i]<-progr.lme(fitlx,model=modeltype,X=T,x=newdata$x[i], time=5,
logbase=invboxcox(x=newdata$nbase[i]))$prob.progr
}

# Step 5: Calculate the integrated sensitivity of the new model
ISnew_lme<-mean(problmex[newdata$progr==1],na.rm=T)

# Step 6: Calculate one minus integrated specifity of the new model
IPnew_lme<-mean(problmex[newdata$progr==0],na.rm=T)

# Step 7: Calculate the IDI
IDI_lme<-(ISnew_lme-IPnew_lme)-(ISold_lme-IPold_lme)
```

Table 20 gives the description of the IDI components as described by Pencina et al (2008) and Table 21 the explanation why the IDI of the longitudinal model is 0.0007 and the IDI of the logistic model is -0.0018 in the scenario above (Figure 18).

Table 20. Description of the IDI components

Abbrev.	Description
ISnew	integral of sensitivity (IS) of the model with the marker x
IPnew	integral of one minus specificity(IP) of the model with the marker x
ISold	IS of the model without the marker x
IPold	IP of the model without the marker x

Table 21. Example of IDI composition; γ_{11}= 0

Description	Logistic model	Longitudinal model
ISnew	0.1981	0.1764
-IPnew	0.1391	0.1062
-ISold	0.1997	0.1753
+IPold	0.1389	0.1058
IDI	**-0.0018**	**0.0007**

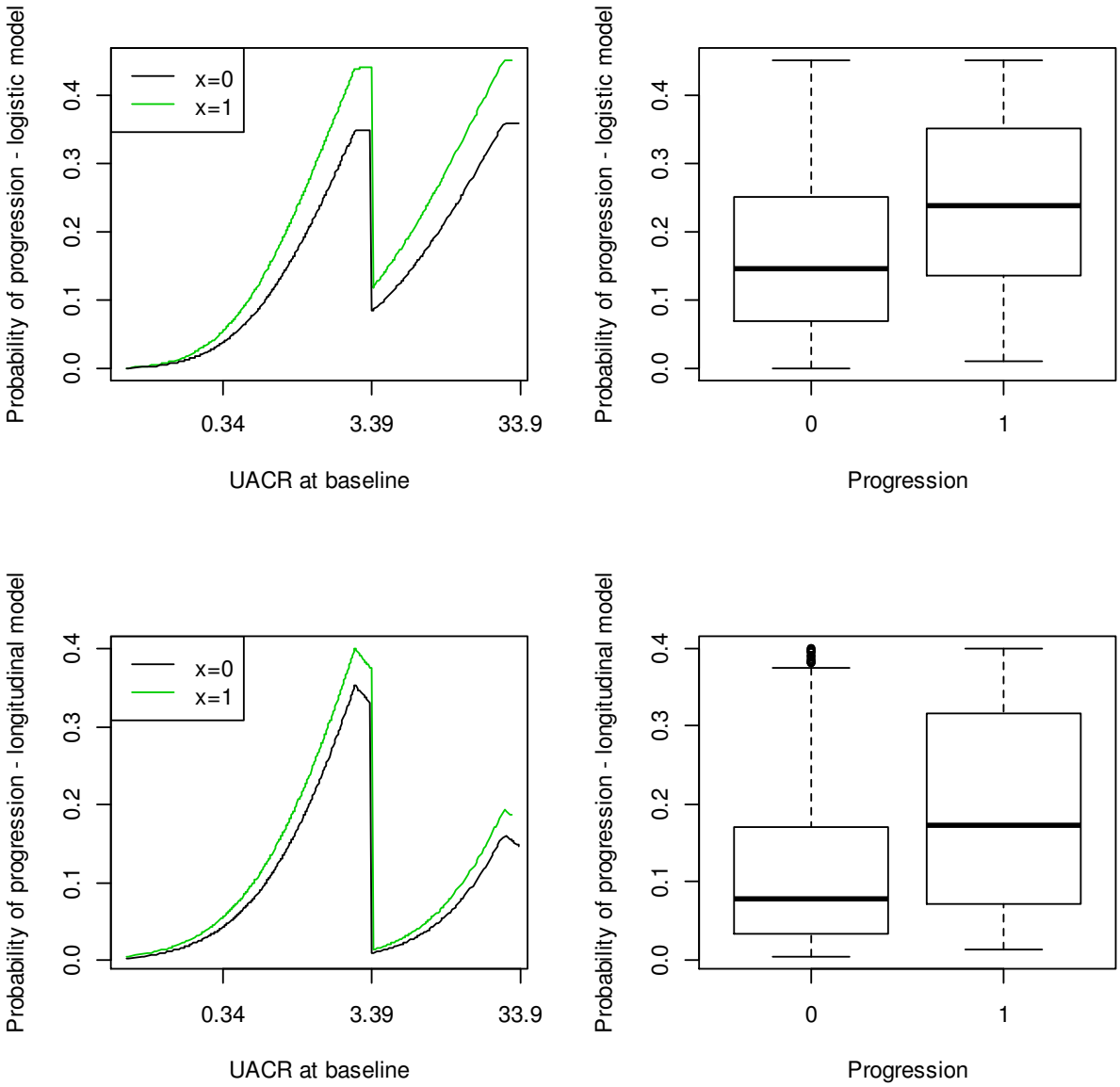
In the scenario from above, the improvement in reclassification is in each, the logistic model and longitudinal model, 18 % (NRI). The following R code shows the calculation of the NRI in the longitudinal model. `Problmex` and `problme` denote the probability of progression estimated by the longitudinal model with and without the risk factor x, respectively.

```
comp<-problmex>problme
events<-(sdata$newdata$progr==1)
nonevents<-(sdata$newdata$progr==0)
nri<-sum(comp[events])/sum(events)-sum(1-comp[events])/sum(events)-
(sum(1-comp[nonevents])/sum(nonevents)-sum(comp[nonevents])/sum(nonevents))
```

Figure 19 illustrates the probability of progression estimated by the logistic and longitudinal LOG model from the simulation in which γ_{11} equals 0.05. In the left panels, the probability of progression is shown for both groups of x and simulated baseline UACR ranging from 0 to 33.9. In the right panels, the probability of progression is visualized in boxplots for patients without progression (progression = 0) and with a progression in CKD (progression = 1).

The probability of progression is, in both models (especially in the logistic model), higher for patients of group $x = 1$ than for patients of group $x = 0$. Furthermore, the boxplots visualize that the estimation of the probability of progression is not perfect since the two boxes overlap to a large degree. This is the case for both the logistic and the longitudinal models.

Figure 19.Probability of progression: in dependence of UACR at baseline (left panel) and progression (right panel); estimated by the logistic LOG model (top panel) and by the longitudinal LOG model(bottom panel); γ_{11}=0.05; seed = 3432; trainings sample no. = 346



In the setting of Figure 19, the IDI for the longitudinal model is 0.0063 and the NRI is 0.289, and the IDI for the logistic model is 0.0116 and the NRI is again 0.289. While the IDI showed that the usefulness of the new marker x is in the longitudinal model better than in the logistic model, the improvement of reclassification is the same in both models with 29 % (NRI). Table 22 contains the intermediate results of the IDI calculation.

Table 22. Example of IDI composition; γ_{11}= 0.05

Description	Logistic model	Longitudinal model
ISnew	0.2469	0.1928
-IPnew	0.1672	0.1167
-ISold	0.2370	0.1885
+IPold	0.1688	0.1187
IDI	**0.0116**	**0.0063**

In summary, all measures (c-index, correlation of predicted probabilities of progression and observed progression status, IDI, NRI) indicate the risk factor x seems to explain little in addition to the baseline urine albumin excretion of a patient; this is already clearly revealed by plotting the predicted progression probabilities. There is large variation in the estimated progression probabilities caused by the baseline UACR, but only little difference in the curves estimated for the groups $x = 0$ and $x = 1$. As the longitudinal model is closer to the data generation procedure, the outcome is already better predicted by the baseline UACR and the risk factor x has even less additional value than in the logistic model, which is misspecified.

4.2.2.2 Optimization of BC parameters

In subsection 4.2.2.1 we have evaluated the logistic and longitudinal model on the basis of simulated patients using the global Box-Cox parameters. By contrast, in this subsection we optimize the Box-Cox parameters on each of the 1000 simulated datasets. These Box-Cox parameters are referred to as the locally optimized or, in short, the 'local Box-Cox parameters'. As in the subsection before, the logistic and longitudinal model are compared to each other with regard to their power to detect the effect of x and their estimated probabilities of progression, as well as on their c-statistics and IDI. As the optimization of the Box-Cox parameters only concerns the BCLOG models, we will focus in this subsection only on this models. For the results of the LOG models return to subsection 4.2.2.1.

The set up has not changed: All coefficients were assumed to be constantly zero (γ_{00} = 0, γ_{01} = 0, γ_{10} = 0) and only γ_{11}, the slope of group $x = 1$, was set to 0.02 and 0.05. For each γ_{11} 1000 simulated datasets of 1041 fictive patients each were generated and evaluated in the same way as before. The seed was always set to 49303.

Table 23. Progression rates and power; n = 1041

γ_{11}	BC optimization	Apparent progression rates*		Apparent relative risk	Power	
		Group x = 0	Group x = 1		Logistic	Longitudinal
0.02	yes	0.180 (0.017)	0.201 (0.018)	1.127	0.149	0.396
0.02	no	0.179 (0.017)	0.203 (0.018)	1.145	0.171	0.386
0.05	yes	0.180 (0.017)	0.247 (0.019)	1.385	0.768	0.984
0.05	no	0.179 (0.017)	0.248 (0.019)	1.399	0.806	0.985

*mean (sd) over 1000 simulated datasets

Table 23 displays that for $\gamma_{11} = 0.02$ the apparent relative risk is slightly lower (1.145 and 1.127, respectively), when the local Box-Cox parameters are used instead of the global. Keep in mind here that the Box-Cox parameters were not optimized on the same datasets. All calculations were executed once for simulated patients using global Box-Cox parameters and once for other simulated patients using local Box-Cox parameters in the log UACR transformations. Furthermore, for $\gamma_{11} = 0.02$ the power of the risk factor for progression in the logistic model dropped to some extent from approx. 17 % to 15 %. In contrast, the power of the risk factor for progression in the longitudinal model increased slightly from approx. 39 % to 40 %.

For $\gamma_{11} = 0.05$ we observe similar changes: The apparent relative risk is to some extent lower (1.399 and 1.385, respectively), when using the local Box-Cox parameters in the data generation process. Moreover, the power of the risk factor for progression in the logistic model dropped from approx. 81 % to 77 %, when applying the optimized Box-Cox parameters.

Table 24. Probabilities of Progression; n = 1041

γ_{11}	BC opt	Estimated probability of progression in the population*			
		Logistic models		Longitudinal models	
		group $x = 0$	group $x = 1$	group $x = 0$	group $x = 1$
0.02	yes	0.176 (0.164, 0.188)	0.196 (0.184, 0.209)	0.120 (0.058, 0.216)	0.130 (0.064, 0.230)
0.02	no	0.179 (0.168, 0.190)	0.203 (0.191, 0.214)	0.136 (0.130,0.142)	0.145 (0.138,0.151)
0.05	yes	0.176 (0.165, 0.188)	0.242 (0.228, 0.256)	0.121 (0.058, 0.215)	0.146 (0.074, 0.249)
0.05	no	0.179 (0.168, 0.190)	0.247 (0.235, 0.260)	0.136 (0.130,0.142)	0.160 (0.152,0.166)

* median (1st and 3rd Quartile) over 1000 simulated datasets

Comparing the probability of progression in the population estimated by the logistic BCLOG model from the simulation, in which global (BC opt = no) or local Box-Cox parameters (BC opt = yes) were used in the transformation, in Table 24 results only in slight decreases in the first position after decimal point. For example, the median (1st and 3rd Quartile) probability of progression over 1000 simulations estimated by the logistic BCLOG model for patients, who belong to group $x = 1$, and where

γ_{11} was set to 0.05, is 24.2 % (22.8 %, 25.6 %) when using the locally optimized Box-Cox parameters and 24.7 % (23.5 %, 26.0 %) when using the global Box-Cox parameters.

The probability of progression by the longitudinal BCLOG models is highly affected by whether the Box-Cox parameters are optimized or not. In group $x = 1$ the estimated median (1st and 3rd Quartile) of the probability of progression over 1000 simulations estimated by the longitudinal model, where γ_{11} = 0.05, is 14.6 % (7.4 %, 24.9 %) using local Box-Cox parameters and 16 % (15.2 %, 16.6 %) when applying global Box-Cox parameters. Notice that the fluctuation range of the probability of progression estimation is much higher when optimizing the Box-Cox parameters on each dataset.

Table 25. True probability of progression; n = 1041

γ_{11}	BC optimization	True probability of progression* x = 0	x = 1	Expected RR	Expected OR
0.02	yes	0.138 (0.137, 0.141)	0.149 (0.147, 0.151)	1.090	1.104
0.02	no	0.139 (0.137, 0.141)	0.149 (0.147, 0.151)	1.090	1.104
0.05	yes	0.138 (0.137, 0.141)	0.166 (0.163, 0.168)	1.237	1.279
0.05	no	0.139 (0.137, 0.141)	0.166 (0.163, 0.168)	1.237	1.279

* median (1st and 3rd Quartile) over 1000 simulated datasets

Regarding the true probability of progression, the expected relative risk and the expected odds ratio the simulations, in which either global or local Box-Cox parameters were used, give almost identical results.

4.2.2.2.1 Optimal Box-Cox parameters

Tables 26 and 27 give a summary of the median (1st and 3rd Quartile) local Box-Cox parameters, which were obtained in 1000 simulated datasets. Recall that the global Box-Cox parameters are $\lambda = 0.5011719, c = 2.8494141$, mean BC log UACR = 1.494655 and standard deviation BC log UACR = 0.8691764. The standard deviation of the difference of the quantiles of a normal distribution and the quantiles of the residuals of the longitudinal model with only time as fixed and random effect was called 'sdOPT'. The parameter sdOPT was minimized in the optimization with the R built-in function `optim`. In other to get a better insight in this optimization an extract of the R Code follows. The self-written R-functions `boxcox2` and `objective` are explained in the appendix.

```
# Step 1: assign start values for Box-Cox parameters
startvalues<-c(1,3)

# Step 2: load R-functions boxcox2 and objective

# Step 3: load R-package nlme
library(nlme)
```

```
# Step 4: optimize the objective function
result<-optim(startvalues, objective, ret=F, x=longi[,"logalb"],
time=longi[,"time"], ID=longi[,"id"])

# Step 5: extract Box-Cox parameters from results
result$par
```

Table 26. Local Box-Cox parameters; n = 1041

γ_{11}	λ^*	c^*
0.02	0.519 (0.495, 0.546)	2.733 (2.642, 2.797)
0.05	0.523 (0.496, 0.551)	2.725 (2.635, 2.792)
true	**0.501**	**2.849**

* median (1st and 3rd Quartile) over 1000 simulated datasets

Table 27. BC log UACR values and standard error of optimization; n = 1041

γ_{11}	Mean BC log UACR*	SD BC log UACR*	sdOPT*
0.02	1.501 (1.467, 1.541)	0.895 (0.865, 0.921)	0 (0, 0)
0.05	1.535 (1.500, 1.576)	0.904 (0.871, 0.931)	0 (0, 0)
true	**1.495**	**0.869**	

* median (1st and 3rd Quartile) over 1000 simulated datasets

In Tables 26 and 27 we see that the local Box-Cox parameters were almost the same in the simulation with $\gamma_{11} = 0.02$ and $\gamma_{11} = 0.05$. Comparing the local Box-Cox parameters to the global Box-Cox parameters ('true') shows that the parameter c is underestimated. The global Box-Cox parameter c is higher than the third quartile of the local Box-Cox parameters c over 1000 simulated data sets.

4.2.2.2.2 Assessing model improvement

As before, c-statistics and integrated discrimination improvement were used for assessing the model improvements. The self-written R functions for the c-statistic and the IDI are explained in the appendix. Notice that all models were calculated twice in the simulation, once without the marker x and once with the marker x. This was a preparation for the calculation of the IDI.

4.2.2.2.2.1 C-statistics

Table 28 compares the c-statistic and the correlation between progression and its estimation by the logistic model to the c-statistic and to the correlation between progression and its estimation by the longitudinal model. This was achieved through 1000 simulation runs for $\gamma_{11} = 0.02$ and $\gamma_{11} = 0.05$, using the fit objects of the logistic and the longitudinal models, which were generated in the previous simulation. The seed was set to 4239.

Table 28. C-statistics and correlation of all models; n = 1041

γ_{11}	BC	C-Statistic*		Correlation*†	
	opt	Logistic model	Longitudinal model	Logistic model	Longitudinal model
0.02	yes	0.673 (0.020)	0.694 (0.020)	0.237 (0.029)	0.263 (0.030)
0.02	no	0.674 (0.021)	0.695 (0.020)	0.238 (0.029)	0.264 (0.031)
0.05	yes	0.678 (0.019)	0.694 (0.020)	0.255 (0.029)	0.276 (0.030)
0.05	no	0.679 (0.020)	0.695 (0.019)	0.257 (0.029)	0.277 (0.031)

* values are given by mean (sd)

† of predicted probabilities of progression and observed progression status

When we compare in Table 28 the c-statistics from simulation with the global Box-Cox parameters (BC opt = no) to those of the simulation with the local Box-Cox parameters (BC opt = yes) we only see a minor difference: The third position after decimal point in the mean c-statistics is slightly lower, when the local Box-Cox parameters are used in the transformation of the simulated log UACR instead of the global. For example, the mean (sd) c-statistic over 1000 simulation runs for the logistic BCLOG model and where γ_{11} was set to 0.05 in the simulation, is 0.679 (0.020) in the simulation with global Box-Cox parameters and 0.678 (0.019) in the simulation, in which the Box-Cox parameters were optimized on each dataset.

Comparing the correlation of predicted probabilities of progression and observed progression status from the simulations, in which the Box-Cox parameters were optimized on each dataset or not, results in only observing non mentionable differences. Again, the correlation of predicted probabilities of progression and observed progression status is slightly higher in the longitudinal models than in the logistic models.

4.2.2.2.2.2 Integrated Discrimination Improvement

We see in Table 29 that the IDI of both models, i.e. the logistic and the longitudinal, is higher for $\gamma_{11} = 0.05$ than for $\gamma_{11} = 0.02$. Furthermore, the IDI of the logistic models is always higher as the IDI in the longitudinal models. Similar results were observed in the simulation, in which the Box-Cox parameters were not optimized on each dataset.

We further observe that the fourth position after decimal point of the mean (sd) IDI is lower when applying local Box-Cox parameters to the transformation of the simulated log UACR instead of the global. For example, if $\gamma_{11} = 0.05$, then the mean (sd) IDI of the logistic BCLOG model is 0.0079 (0.0046) in the simulation with global Box-Cox parameters and 0.0073 (0.0043) in the simulation with local Box-Cox parameters.

Table 29. IDI of all models; n = 1041

γ_{11}	BC optimization	IDI*†	
		Logistic model	Longitudinal model
0.02	yes	0.0008 (0.0019)	0.0004 (0.0006)
0.02	no	0.0011 (0.0020)	0.0004 (0.0007)
0.05	yes	0.0073 (0.0043)	0.0026 (0.0013)
0.05	no	0.0079 (0.0046)	0.0027 (0.0015)

* values are given by mean (sd) over 1000 simulated datasets
† of the logistic and longitudinal model for the marker x

The differences in the logistic and longitudinal models are all significant for both, the mean c-statistics and mean IDI, when applying a two-sided t-test.

4.3 Analysis of ONTARGET-SysKid study

This subsection is split into two main parts: a descriptive analysis of the ONTARGET-SysKid dataset and the comparison of logistic and longitudinal models, which were evaluated on this dataset. Patients with UACR measurements at baseline and at the 2 and 5 years visits were included in the dataset ($n = 5293$). This means, we consider only complete cases here.

The randomization groups for the double-blind treatment of the patients are telmisartan (comparison group, i.e. $x_1 = 0$ and $x_2 = 0; n = 1784$), ramipril ($x_1 = 1; n = 1734$) and the combination treatment (ramipril and telmisartan; $x_2 = 1; n = 1775$). In the following these groups will be referred to as 'Telmisartan', 'Ramipril' and 'Combination'.

4.3.1 Descriptive analysis

In order to get a general idea of the urine albumin excretion values of the patients from the ONTARGET-SysKid dataset, some descriptive Tables and Figures are given below.

Table 30 shows the 10 %, 50 % and 90 % percentiles of UACR from these patients with either normoalbuminuria or microalbuminuria at baseline.

Table 30. Percentiles (10%, 50% and 90%) of UACR at baseline; n = 5293

Percentiles of UACR*	10 %	50 %	90 %
patients with normoalbuminuria	0.168	0.498	1.943
patients with microalbuminuria	4.000	8.381	23.781

*UACR is measured in mg/mmol.

Table 31 gives an overview of some descriptive measures concerning the urine albumin excretion values of the patients, which were measured at baseline and at the 2 and 5 years visits.

Table 31. Descriptive Statistics of UACR; n = 5293

UACR*	Baseline	2 years	5 years
minimum	0.045	0.064	0.055
1st quartile	0.307	0.339	0.367
median	0.676	0.760	0.943
mean	2.841	5.258	7.653
3rd quartile	2.333	2.662	3.733
maximum	33.890	336.000	375.400

*measured in mg/mmol

Figure 20 illustrates the distribution of the urine albumin excretion from the above described data-set at each time point of measurement. Please note that the histograms of UACR measured at the 2 and 5 years visits were only plotted from 0 to 100 mg/mmol to guarantee a better visualization. Values above 100 mg/mmol represent only 1 % of the urine albumin excretion values at these visits.

Figure 20.Histograms of UACR measured at baseline, 2 years and 5 years; n = 5293

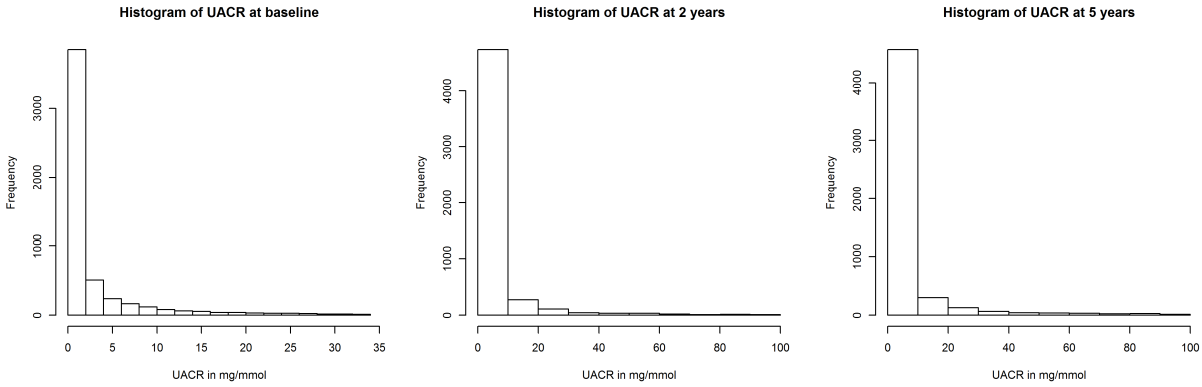


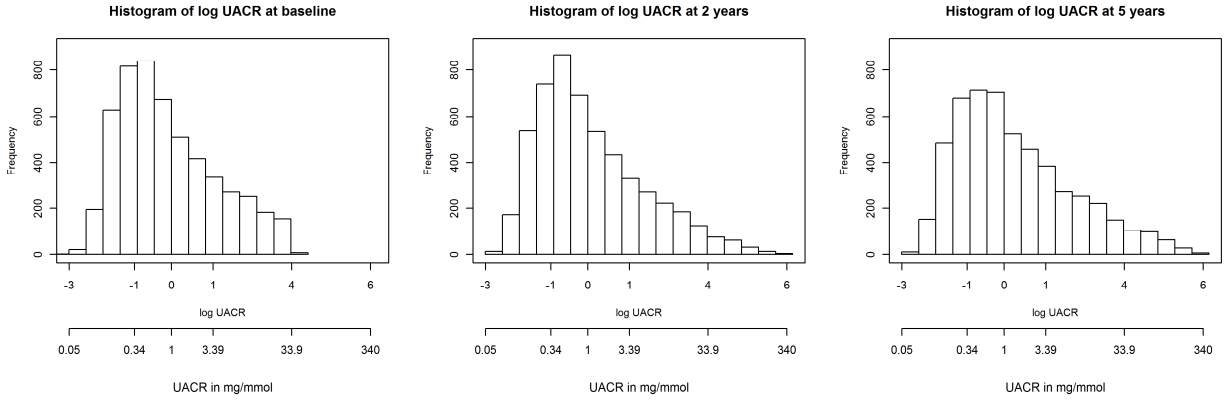
Table 32. Descriptive statistics of log UACR; n = 5293

Log UACR	Baseline	2 years	5 years
minimum	-3.091	-2.749	-2.896
1st quartile	-1.183	-1.082	-1.002
median	-0.392	-0.274	-0.059
mean	-0.084	0.081	0.299
3rd quartile	0.847	0.979	1.317
maximum	3.523	5.817	5.928

Table 32 and Figure 21 demonstrate how the range of the urine albumin excretion values looks like after a log transformation. In particular, it is now easier to distinguish the distributions. For example,

when we compare the distribution of the log UACR at baseline to the one at the 5 years visit, we clearly see an increase in UACR for some patients.

Figure 21. Histograms of log UACR measured at baseline, 2 years and 5 years; n = 5293



In a further step, descriptive statistics of log UACR were calculated for all measurement time points in the three randomization groups and summarized in Table 33. At baseline the distribution of log UACR is more or less the same for each group, clearly, since the patients were randomly assigned to a group. At the 2 years visit the distribution of log UACR shifted more to the right, i.e. to higher values of log UACR, with a median (1st and 3rd quartile) log UACR of -0.250 (-1.011; 1.023) in Ramipril and -0.247 (-1.077; 1.061) in Telmisartan than in Combination with a median (1st and 3rd quartile) log UACR of -0.325 (-1.152; 0.862). At the 5 years visit this effect slightly reinforced with a median (1st and 3rd quartile) log UACR of -0.032 (-0.968; 1.369) in Ramipril and 0.000 (-0.987; 1.342) in Telmisartan than in Combination with a median (1st and 3rd quartile) log UACR of -0.127 (-1.061; 1.190). From this point of view it looks like if the combination treatment worked best in stopping the UACR from increasing. Figures 22 to 24 visualize this effect.

Table 33. Descriptive statistics of log UACR for all randomization groups; n = 5293

log(UACR)	Ramipril			Combination			Telmisartan		
	Baseline	2 years	5 years	Baseline	2 years	5 years	Baseline	2 years	5 years
minimum	-2.792	-2.557	-2.896	-2.830	-2.739	-2.808	-3.091	-2.749	-2.605
1st quartile	-1.230	-1.011	-0.968	-1.173	-1.152	-1.061	-1.134	-1.077	-0.987
median	-0.411	-0.250	-0.032	-0.388	-0.325	-0.127	-0.388	-0.247	0.000
mean	-0.097	0.134	0.361	-0.096	-0.004	0.211	-0.060	0.115	0.327
3rd quartile	0.847	1.023	1.369	0.818	0.862	1.190	0.867	1.061	1.342
maximum	3.502	5.817	5.904	3.516	5.707	5.542	3.523	5.442	5.928

Figure 22. Log UACR measured at baseline for each randomization group (group 1 = Ramipril, group 2 = Combination and group 3 = Telmisartan)

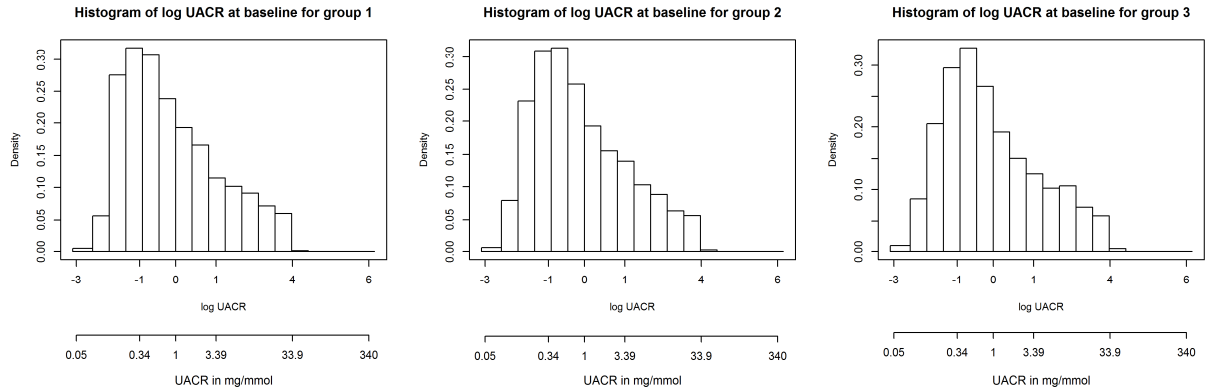


Figure 23. Log UACR measured at the 2 years visit for each randomization group (group 1 = Ramipril, group 2 = Combination and group 3 = Telmisartan)

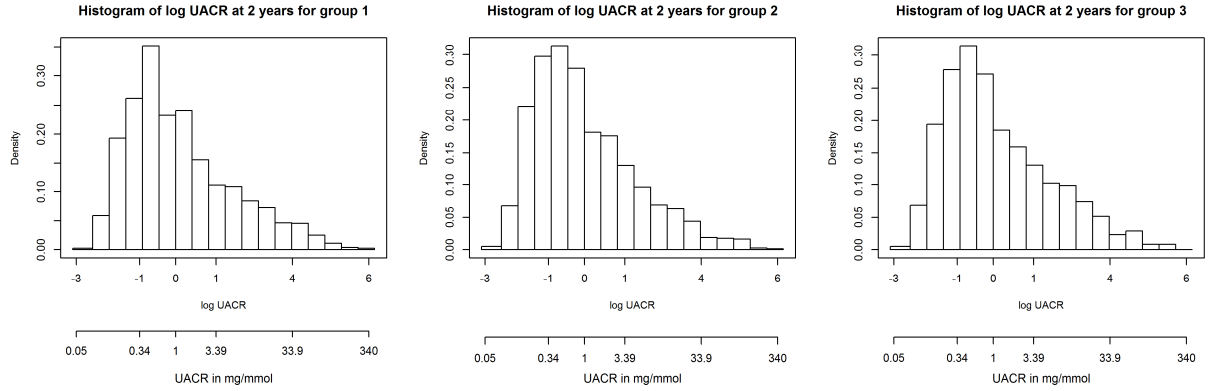
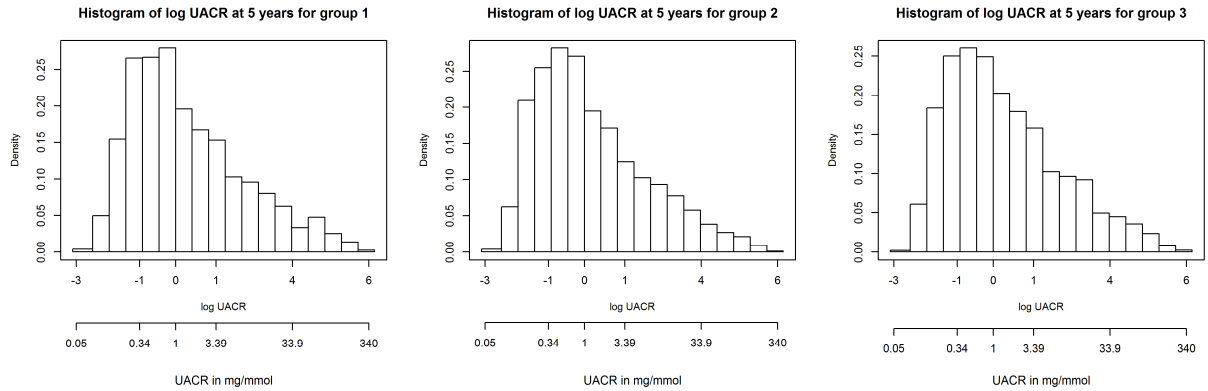


Figure 24. Log UACR measured at the 5 years visit for each randomization group (group 1 = Ramipril, group 2 = Combination and group 3 = Telmisartan)



The next step was to find a suitable Box-Cox transformation to guarantee the assumption of normally distributed residuals in the longitudinal model. In contrast to the Box-Cox transformation which was used in the simulations, this transformation depends only on the parameter λ, because the parameter *c* is highly dependent on its start value and it has furthermore nearly no impact on the log UACR values (see Table 34). Therefore we reduced the Box-Cox transformation to the parameter λ, which required new R functions (see `boxcox3` and `boxcox4` in the appendix).

Table 34. Box-Cox parameters optimization on ONTARGET SysKid dataset; n = 5771

Start values		Final BC Parameter		BC log UACR		sdOPT
λ	c	**λ**	***c***	Mean	SD	
1.000	2.000	0.500	2.492	1.483	0.864	0.051
0.113	0.226	0.500	0.096	1.483	0.869	0.047

The parameter λ was optimized on the ONTARGET-SysKid dataset ($n = 5293$). For different λ, ranging from -1 to 3, the standard deviation of the difference of the quantiles of standard normal distribution and quantiles of the Box-Cox-transformed log urine albumin excretion values of underlying dataset was calculated. The minimum standard deviation was found manually at $\lambda = 0.67$ with a value of 0.053127. For this new Box-Cox transformation the input values were again centralized by the mean and standard deviation of the underlying dataset. The self-written function `target` (see R-functions in the appendix) returned for $\lambda = 0.67$ and the ONTARGET-SysKid dataset a mean BC log UACR of 1.041583 (`mean_boxcox`) and a standard deviation of 1.17939 (`sd_boxcox`).

Table 35 and Figure 25 illustrate the distribution of the Box-Cox log transformed urine albumin excretion values at each time point of measurement.

Table 35. Descriptive statistics of Box-Cox transformed log UACR; n = 5293

BC log UACR	Baseline	2 years	5 years
minimum	-2.149	-2.149	-2.149
1st quartile	-0.801	-0.720	-0.656
median	-0.210	-0.131	0.012
mean	-0.112	-0.010	0.121
3rd quartile	0.569	0.645	0.836
maximum	1.963	2.990	3.038

Figure 25. Histograms of Box-Cox transformed log UACR, which was measured at baseline, 2 years and 5 years

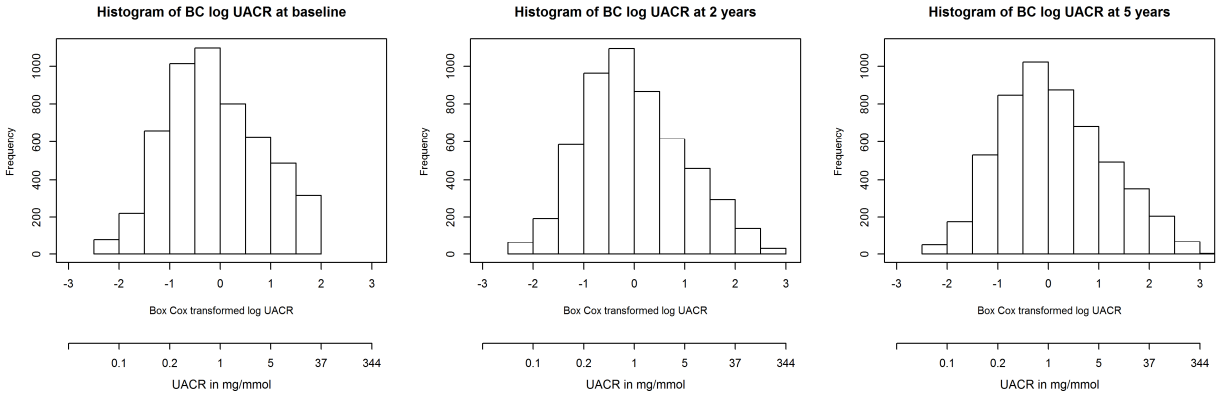


Table 36 and Figures 26 to 28 give a general idea of the distribution of the Box-Cox transformed log UACR for each randomization group at the three time points of measurement.

Table 36. Descriptive statistics of BC log UACR by randomization groups and measurement time points; n = 5293

BC log UACR	Ramipril			Combination			Telmisartan		
	Baseline	2 years	5 years	Baseline	2 years	5 years	Baseline	2 years	5 years
minimum	-2.149	-2.149	-2.149	-2.149	-2.149	-2.149	-2.149	-2.149	-2.149
1st quartile	-0.840	-0.663	-0.629	-0.793	-0.776	-0.702	-0.761	-0.715	-0.645
median	-0.223	-0.114	0.030	-0.208	-0.164	-0.032	-0.208	-0.112	0.050
mean	-0.118	0.025	0.161	-0.119	-0.063	0.065	-0.097	0.009	0.139
3rd quartile	0.568	0.670	0.865	0.552	0.577	0.765	0.580	0.692	0.850
maximum	1.952	2.990	3.027	1.959	2.944	2.873	1.963	2.830	3.038

Figure 26. BC log UACR measured at baseline for group 1 = Ramipril, group 2 = Combination and group 3 = Telmisartan

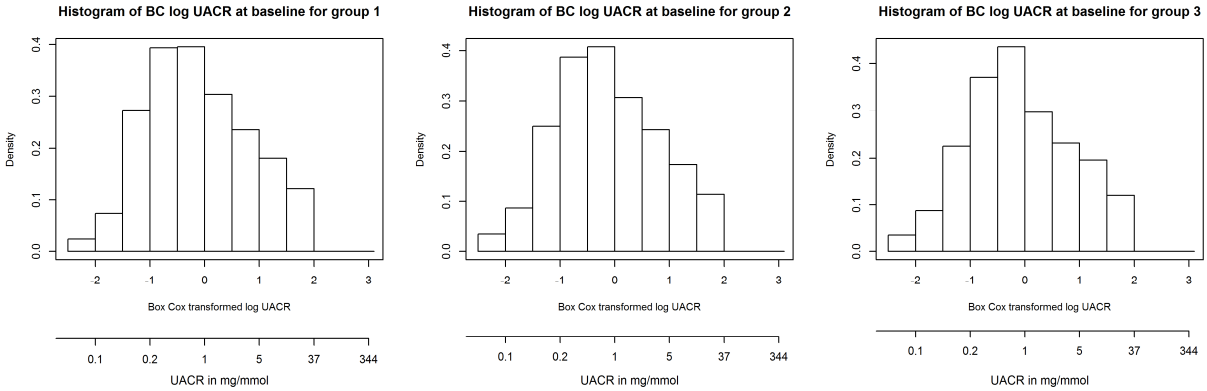


Figure 27. BC log UACR measured at 2 years for group 1 = Ramipril, group 2 = Combination and group 3 = Telmisartan

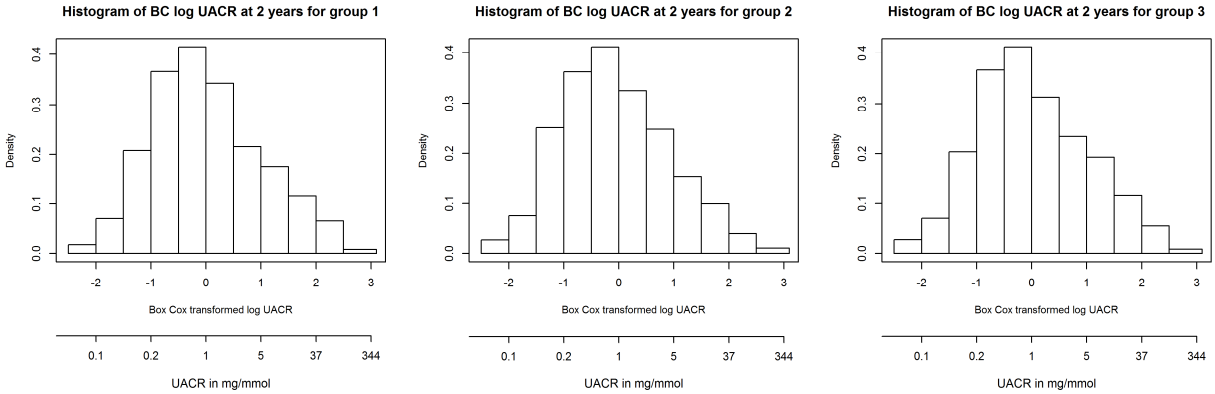
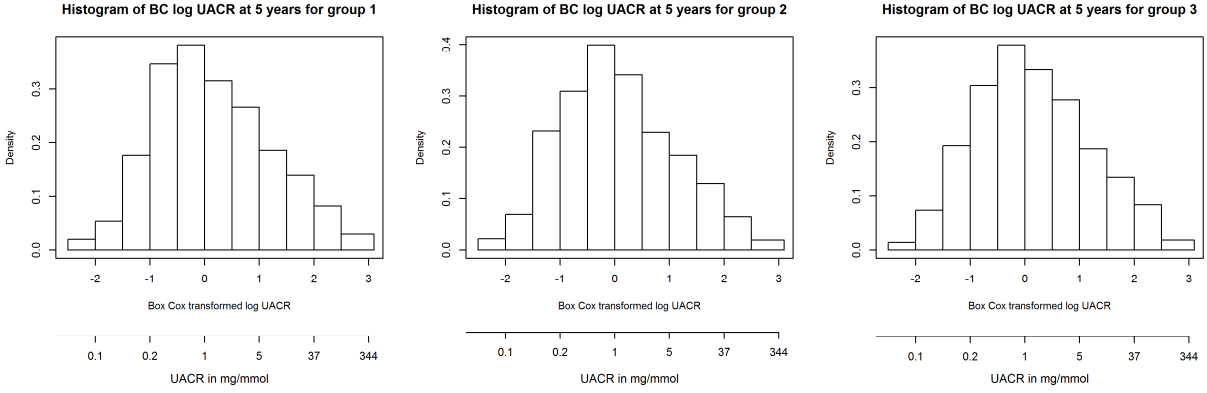


Figure 28. BC log UACR measured at 5 years for group 1 = Ramipril, group 2 = Combination and group 3 = Telmisartan



In a further step, we looked at the apparent progression rates of each randomization group. The progression rates in this dataset were calculated as the percentage of patients who experienced progression. The progression rates are: 18.0 % under Ramipril, 18.0 % under Telmisartan and only 15.7 % under Combination. Thus, the findings of Mann et al (2008: 550f) on the whole ONTARGET study population could be reproduced in the SysKid subpopulation of diabetic patients. Mann et al (2008: 550f) wrote: "The risk of developing new microalbuminuria, macroalbuminuria, or both, during the trial was not different in groups on telmisartan or ramipril (949, 11.1 % vs 1018, 11.7 %; HR 0.94, 0.96-1.02, p=0.119) but was lower with combination therapy than with ramipril (888, 10.4 % vs 1018, 11.7 %; HR 0.88, 0.81-0.96, p=0.03)."

4.3.2 Logistic models

We see in Tables 37 and 38 that the intercept and d are significant (p-value: 0.000) in the logistic BCLOG model and logistic LOG model, while the randomization groups are not significant (p-value > 0.050). In Table 37 we further see that the odds of progression for Ramipril (i.e. $exp(0.007) = 1.007$) are 0.7 % higher than for Telmisartan. The odds of progression for Combination (i.e. $exp(-0.158) = 0.8538$) are 14.6 % lower than for Telmisartan. Furthermore, the effect on the odds of a one-unit increase of d is $exp(-0.958) = 0.3837$, i.e. odds decrease by 61.6 % ($1 - 0.3837$). This is self-evident, since the closer the baseline value of a patient is to a cut-off point, the higher is his/her probability of progression.

Table 37. Logistic BCLOG model: Estimated coefficients; n = 5293

Coefficients	Estimate	Std. Error	z value	p-value
Intercept	-0.560	0.085	-6.561	0.000
Ramipril	0.007	0.090	0.075	0.940
Combination	-0.158	0.092	-1.723	0.085
d	-0.958	0.065	-14.635	0.000

Table 38. Logistic LOG model: Estimated coefficients; n = 5293

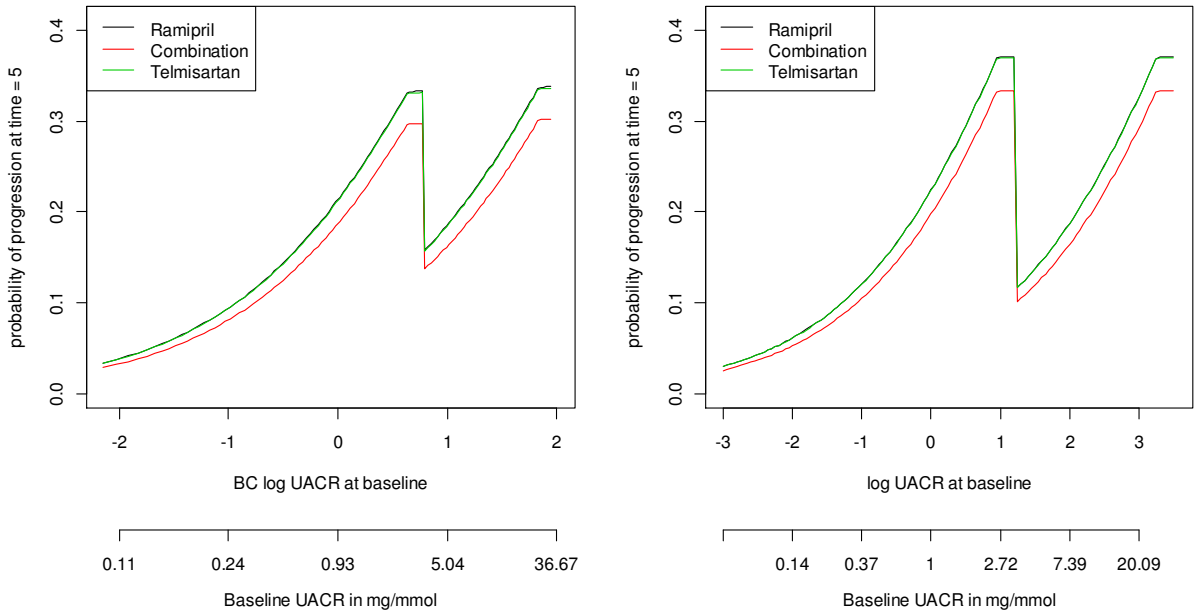
Coefficients	Estimate	Std. Error	z value	p-value
Intercept	-0.337	0.091	-3.699	0.000
Ramipril	0.002	0.090	0.026	0.979
Combination	-0.158	0.092	-1.717	0.086
d	-0.742	0.046	-16.198	0.000

Table 38 shows that the odds of progression for Ramipril (i.e. $exp(0.002) = 1.002$) are 0.2% higher than for Telmisartan. The odds of progression for Combination (i.e. $exp(-0.158) = 0.8538$) are

14.6 % lower than for Telmisartan. The effect on the odds of a one-unit increase of d is $exp(-0.742) = 0.4762$, i.e. the odds decrease by 52.4 % $(1 - 0.4762)$.

Figure 29 visualizes the probability of progression estimated by the logistic models. In the left panel, the estimation using Box-Cox transformed log urine albumin excretion values (BCLOG model) and in the right panel, the same estimation executed with log urine albumin excretion values (LOG model) is shown. We see in line charts that the estimated probability of progression is the same for Ramipril and Telmisartan and that the probability of progression at baseline is always lower in Combination for all log UACR. The difference between the two models is that the misspecified LOG model estimates a slightly higher probability of progression than the BCLOG model.

Figure 29.Probability of progression estimated by logistic models (left panel: BCLOG model and right panel: LOG model)



4.3.3 Longitudinal models

Tables 39 and 40 give an overview of the longitudinal BCLOG and LOG model, respectively. The independent variable time (which either assumes 0, 2 or 5) in both longitudinal models is significant (p-value < 0.001). In the BCLOG model the intercept is significant (p-value: 0.003), while in the LOG model it is not significant (p-value: 0.928). All other parameters are not significant (p-value > 0.050) in both longitudinal models.

In Table 39 we see that the estimated trajectory of the Box-Cox transformed log UACR is $-0.063 + 0.047 \cdot time$ for a patient who belongs to Telmisartan, $-0.063 - 0.01 + 0.047 \cdot time + 0.008 \cdot time = -0.073 + 0.055 \cdot time$ for a Ramipril patient and $-0.063 - 0.04 + 0.047 \cdot time - 0.009 \cdot time = -0.103 + 0.038 \cdot time$ for a Combination patient.

Table 39. Longitudinal BCLOG model: Estimated coefficients; n = 5293

Coefficients	Value	Std.Error	DF	t-value	p-value
Intercept	-0.063	0.021	11032	-2.990	0.003
time	0.047	0.005	11032	10.146	0.000
Ramipril	-0.010	0.030	5739	-0.337	0.736
Combination	-0.040	0.030	5739	-1.370	0.171
time:Ramipril	0.008	0.007	11032	1.251	0.211
time:Combination	-0.009	0.007	11032	-1.446	0.148

Table 40 shows that $-0.003 + 0.078 \cdot time$ is the estimated trajectory of log UACR for a Telmisartan patient, $-0.003 - 0.02 + 0.078 \cdot time + 0.014 \cdot time = -0.023 + 0.092 \cdot time$ for a Ramipril patient and $-0.003 - 0.069 + 0.078 \cdot time - 0.014 \cdot time = -0.072 + 0.064 \cdot time$ for a Combination patient.

Table 40 Longitudinal LOG model: Estimated coefficients; n = 5293

Coefficients	Value	Std.Error	DF	t-value	p-value
Intercept	-0.003	0.032	11032	-0.090	0.928
time	0.078	0.007	11032	10.857	0.000
Ramipril	-0.020	0.046	5739	-0.450	0.653
Combination	-0.069	0.045	5739	-1.536	0.125
time:Ramipril	0.014	0.010	11032	1.368	0.171
time:Combination	-0.014	0.010	11032	-1.427	0.154

Figure 30. Development of log UACR over time estimated by the longitudinal models (left panel: BCLOG model and right panel: LOG model); n = 5293

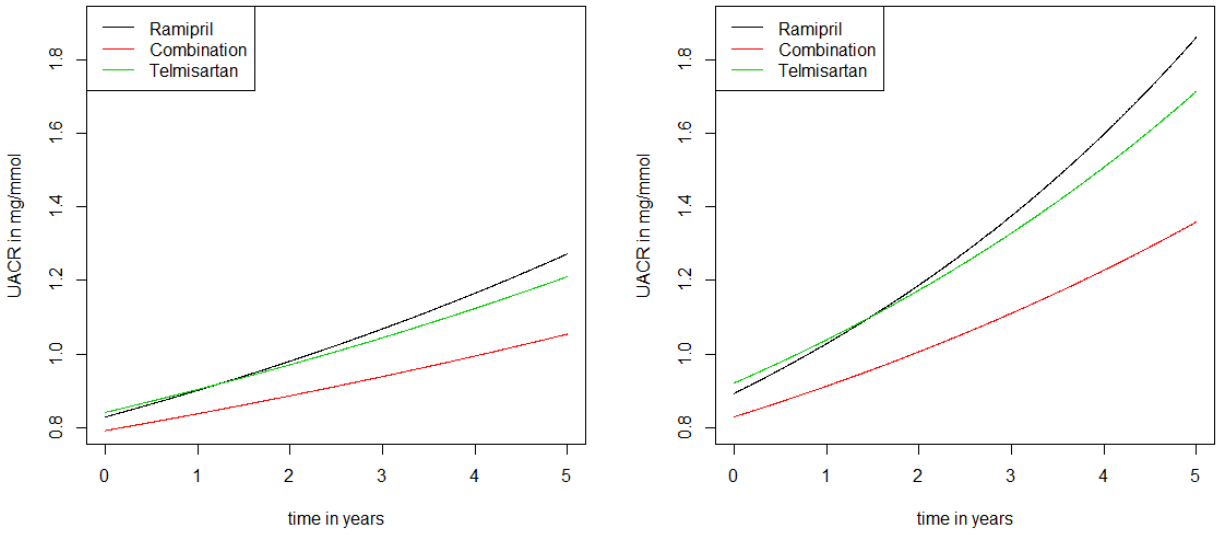


Figure 30 visualizes the development of log urine albumin excretion over time for each randomization group estimated through the longitudinal models (left panel: BCLOG model, right panel: LOG model) as shown in Tables 39 and 40. As we see in these two line charts, UACR increases over time. The slope of UACR is the steepest for Ramipril over time. We further notice that Combination starts with a lower urine albumin excretion at baseline than the other two randomization groups. Moreover, keep in mind that these lines represent only the mean trajectories for each randomization group. The variation between individual UACR trajectories is considerable and not shown in this Figure.

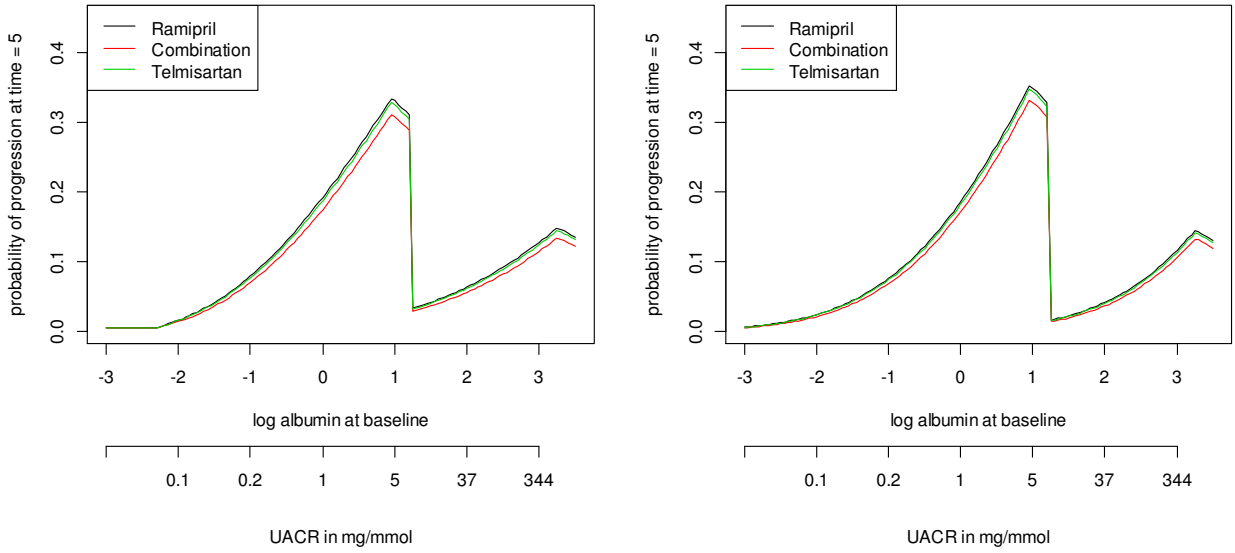
The following R Code shows how the trajectories of UACR estimated by the longitudinal BCLOG model (left panel of Figure 30) were exactly obtained. For that, the UACR of all patients were evaluated on 1000 equally spaced time points between 0 and 5 years. For the longitudinal LOG model, the six steps are analogous.

```
# Step 1: filter model coefficients
b0.u1<-ontarget$tab.fitlx.u1[1,1] #intercept
b1.u1<-ontarget$tab.fitlx.u1[2,1] #time
b2.u1<-ontarget$tab.fitlx.u1[3,1] # Ramipril
b3.u1<-ontarget$tab.fitlx.u1[4,1] # Combination
b4.u1<-ontarget$tab.fitlx.u1[5,1] #time:Ramipril
b5.u1<-ontarget$tab.fitlx.u1[6,1] #time:Combination

# Step 2: define the time interval
time<-seq(0,5,by=0.005)

# Step 3: initialize several variables
BClogRamipril<-BClogCombination<-BClogTelmisartan<-0
logRamipril<-logCombination<-logTelmisartan <-numeric(length(time))

# Step 4: load self-written R function invBCwinsor

# Step 5: calculated BC log UACR for each treatment group in each time
# point and transform them into log UACR
for(i in 1:length(time))
{
       BClogRamipril[i]<-b0.u1+time[i]*b1.u1+b2.u1+b4.u1*time[i]
       logRamipril[i]<-invBCwinsor(x=BClogRamipril[i])

       BClogCombination[i]<- b0.u1+time[i]*b1.u1+b3.u1+b5.u1*time[i]
       logCombinationl[i]<-invBCwinsor(x=BClogCombination[i])

       BClogTelmisartan[i]<- b0.u1+time[i]*b1.u1
       logTelmisartan[i]<-invBCwinsor(x=BClogTelmisartan[i])
}
```

```
# Step 6: plot trajectories in UACR
plot(time,exp(logRamipril), type="l",xlab="time in years",
ylab="UACR in mg/mmol", ylim=c(0.8,1.9))
lines(time, exp(logCombination), col=2)
lines(time, exp(logTelmisartan), col=3)
legend("topleft", c("Ramipril","Combination","Telmisartan"),col=1:3,lwd=1)
```

Figure 31 illustrates the probability of progression for the randomization groups estimated by the longitudinal models at time point 5 years after study enrollment (left panel: BCLOG model, right panel: LOG model). Like in the logistic models, the probability of progression estimated by the longitudinal models is the same for Ramipril and Telmisartan and a little lower for Combination.

Figure 31.Probability of progression at time = 5 for the randomization groups estimated by the longitudinal models (left panel: BCLOG model and right panel: LOG model); n = 5293



4.3.4 Comparison of both models

In a further step the two models of interest, i.e. the logistic and longitudinal models, were compared to each other with regard to their estimated progression probabilities. Table 41 shows the probability of progression estimated by the logistic and longitudinal models for the three randomization groups for six typical patients with a baseline UACR at the 10th, 50th and 90th percentiles within the normoalbuminuria and microalbuminuria classes. Recall that the baseline values of UACR for these patients are given in Table 30. We see that for each randomization group and all typical patients the probability of progression estimated by the logistic model is always higher than the estimation by the longitudinal model. The only exception is for Combination patients with normoalbuminuria (at the 90 % percentile of UACR). In that case, the probability of progression is with 26.40 % slightly lower in the logistic BCLOG model than in the longitudinal BCLOG model with 26.78 %.

Furthermore, as Figures 29 and 31 already visualized, the estimated probability of progression in Combination is always lower than in the other two treatment groups. This is the case for all models and all typical patients.

Table 41. All models: Probability of progression estimated for 6 typical patients in all randomization groups

Model	Albuminuria class at baseline ('typical patients')		Probability of progression in ...		
			Ramipril	Combination	Telmisartan
BCLOG logistic	normo-	10%	0.0688	0.0590	0.0684
		50%	0.1529	0.1328	0.1521
		90%	0.2971	0.2640	0.2957
	micro-	10%	0.1684	0.1467	0.1675
		50%	0.2284	0.2007	0.2272
		90%	0.3282	0.2930	0.3267
LOG logistic	normo-	10%	0.0713	0.0614	0.0712
		50%	0.1470	0.1280	0.1467
		90%	0.3214	0.2875	0.3209
	micro-	10%	0.1278	0.1098	0.1275
		50%	0.2024	0.1777	0.2020
		90%	0.3549	0.3191	0.3544
BCLOG longitudinal	normo-	10%	0.0258	0.0222	0.0248
		50%	0.1086	0.0971	0.1055
		90%	0.2892	0.2678	0.2835
	micro-	10%	0.0379	0.0329	0.0365
		50%	0.0703	0.0621	0.0681
		90%	0.1414	0.1276	0.1377
LOG longitudinal	normo-	10%	0.0317	0.0279	0.0307
		50%	0.1023	0.0926	0.0998
		90%	0.2973	0.2781	0.2923
	micro-	10%	0.0195	0.0170	0.0188
		50%	0.0479	0.0426	0.0465
		90%	0.1353	0.1234	0.1322

Figure 32 illustrates the probability of progression estimated by the logistic BCLOG model (left panel) and by the longitudinal BCLOG model (right panel) for patients with progression in CKD (progression = 1) or not (progression = 0). The boxplots visualize that the estimation of the probability of progression is not perfect, since the boxes overlap to a large degree. This is the case for both models, the logistic and the longitudinal.

Figure 32.Probability of progression estimated by logistic BCLOG model (left panel) and longitudinal BCLOG model (right panel); n = 5293

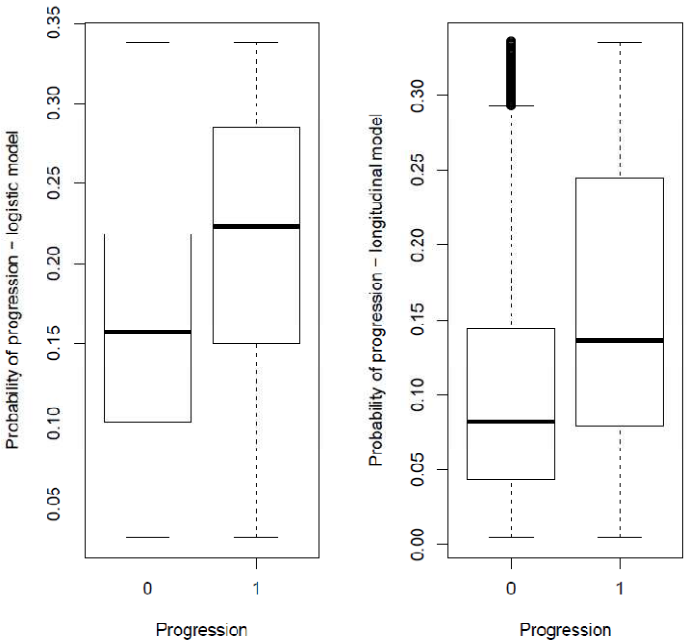


Figure 33. Probability of progression estimated by logistic LOG model (left panel) and longitudinal LOG model (right panel); n = 5293

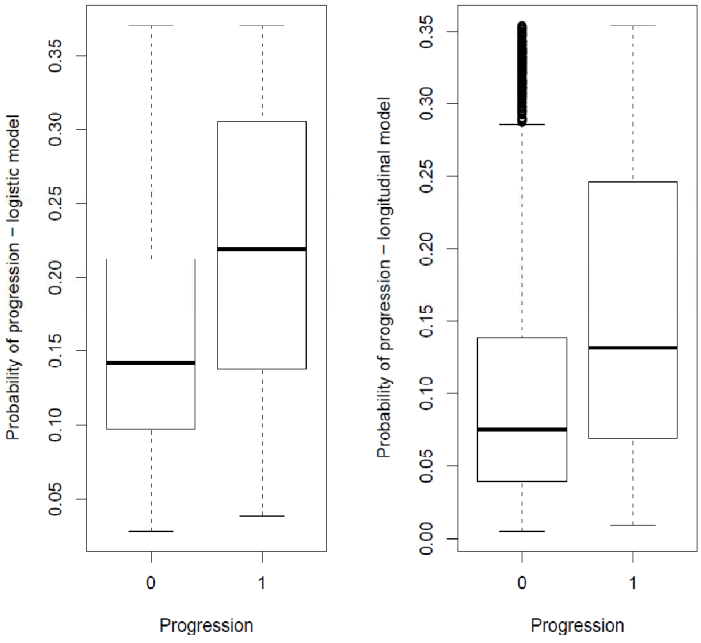


Figure 33 illustrates the probability of progression estimated by the logistic LOG model (left panel) and by the longitudinal LOG model (right panel) for patients with progression in CKD (progression = 1) or not (progression = 0).

It is not surprising that the estimated progression probabilities from both models, i.e. the logistic and the longitudinal, and from both transformations (BCLOG and LOG), are higher for patients who ac-

tually experienced progression than for those who did not experience progression in CKD during the first five years after enrollment, as shown in Figures 32 and 33. Furthermore, we see that the logistic model estimates a higher probability of progression than the longitudinal model.

4.3.5 Analysis of sensitivity

We conducted an analysis of sensitivity on the ONTARGET-SysKid dataset (n = 5293). For this analysis we evaluated the logistic models once more. This time, the binary outcome variable 'renal.event' included in addition to the events 'incidence' and 'progression' the renal events 'doubling of serum creatinine' and 'incidence of end stage renal disease' (ESRD). If renal.event = 0, then the patient has either no renal outcome ($n = 4275$, 80.77 %) or is dead ($n = 58$, 1.10 %) and renal.event = 1 indicates that the patient has a renal outcome ($n = 960$, 18.14 %). ESRD was defined by Dunkler et al (2013) as a glomerular filtration rate less than 15 mL/min/1.73 m² or renal replacement therapy for more than 2 months.

Table 42 shows the probability of a renal outcome estimated by the logistic models for each randomization group. Like before in the scenario with only two renal events, incidence and progression, also the probability of this renal outcome is lowest in the randomization group Combination.

Table 42. Logistic models: Probability of a renal outcome for each randomization group

model	probability of a renal outcome*		
	Ramipril	Combination	Telmisartan
logistic BCLOG	0.185 (0.149, 0.255)	0.170 (0.137, 0.236)	0.191 (0.154, 0.262)
logistic LOG	0.165 (0.128, 0.266)	0.152 (0.118, 0.247)	0.170 (0.133, 0.273)

* median (1st and 3rd quartile)

Tables 43 and 44 give the estimated coefficients of the logistic BCLOG and LOG model, respectively, using the composite endpoint as outcome variable. Comparing the models with all renal events as outcome variable (Tables 43 and 44) to the models with only incidence or progression as outcome variable (Tables 37 and 38), reveals that the models are quite similar. In all logistic models d is significant (p-value: 0.000) and the effect of randomization groups are not significant (p-value > 0.050). As before the variable d expresses the difference between the UACR at baseline and the cut-off point for declaring progression given that baseline UACR. Depending on the scale either the BC log UACR at baseline for the BCLOG model or the log UACR at baseline for the LOG model is used for the definition of d. When comparing the estimated coefficients we notice the following two things. First, a reversed relationship: The odds of progression for Ramipril (i.e. $exp(-0.034) = 0.967$) are 3.3 % lower than for Telmisartan in the logistic BCLOG model with the composite endpoint as outcome variable, while in the logistic BCLOG model with only incidence and progression as outcome, we no-

ticed the opposite relation with odds at 1.007. And second, a similar relation: The odds of a renal event for Combination (i.e. $exp(-0.137) = 0.872$) are 12.6 % lower than for Telmisartan in the logistic BCLOG model with the composite endpoint. In the logistic BCLOG model with only incidence or progression as outcome the odds of incidence or progression for Combination are 14.6 % lower than for Telmisartan. Furthermore, the effect on the odds of a one-unit increase of d is $exp(-0.869) = 0.419$, i.e. the odds decrease by 58.1 % $(1 - 0.419)$ in the logistic BCLOG model with the composite endpoint as outcome variable, which is very similar to the decrease by 61.6 % in the logistic BCLOG model with only incidence and progression as outcome.

Table 43.Logistic BCLOG model with a composite endpoint: Estimated coefficients; n = 5293

Coefficients	Estimate	Std. Error	z value	p-value
Intercept	-0.682	0.082	-8.273	0.000
Ramipril	-0.034	0.087	-0.391	0.696
Combination	-0.137	0.088	-1.560	0.119
d	-0.869	0.063	-13.887	0.000

Table 44 shows that the odds of a renal event for Ramipril (i.e. $exp(-0.038) = 0.963$) are 3.7 % lower than for Telmisartan. The odds of a renal event for Combination (i.e. $exp(-0.137) = 0.872$) are 12.8 % lower than for Telmisartan. The effect on the odds of a one-unit increase of d is $exp(-0.674) = 0.510$, i.e. the odds decrease by 49 %.

Table 44.Llogistic LOG model with a composite endpoint: Estimated coefficients; n = 5293

Coefficients	Estimate	Std. Error	z value	p-value
Intercept	-0.477	0.088	-5.426	0.000
Ramipril	-0.038	0.088	-0.437	0.662
Combination	-0.137	0.088	-1.551	0.121
d	-0.674	0.044	-15.345	0.000

4.3.6 Assessment of model improvement

Finally, the model improvement was assessed using c-statistics, correlation of predicted probabilities of progression and observed progression status and IDI. The new marker, which is required for the calculation of the IDI, was defined as the randomization groups. Therefore, logistic and longitudinal models with and without the randomization groups as dependent variables were computed.

Table 45 gives an overview of the measures of model improvement for each investigated model on the ONTARGET-SysKid dataset. Notice that the outcome variable of the logistic models is again defined by the two renal events, incidence or progression, only.

Table 45. Assessment of model improvement of the logistic and longitudinal models; n = 5293

Measures	**Logistic**		**Longitudinal**	
	BCLOG model	LOG model	BCLOG model	LOG model
c-statistics	0.6692	0.6743	0.6633	0.6609
correlation*	0.2320	0.2474	0.2279	0.2269
IDI	0.0006	0.0005	0.0002	0.0001

* of predicted probabilities of progression and observed progression status

Table 45 basically shows that there is no noticeable difference between the logistic and the longitudinal models regarding their model improvement measures. The c-statistic in the logistic BCLOG model is 0.669 while in the longitudinal BCLOG model it is 0.663, the correlation predicted and observed progression is 0.232 in the logistic BCLOG model and 0.230 in the longitudinal BCLOG model and finally, the IDI in the logistic BCLOG model is 0.0006 and 0.0002 in the longitudinal BCLOG model. In the LOG models we find similar results. Therefore, we conclude that the model improvement measures are only slightly higher in the logistic models than in the longitudinal.

5 Conclusions

The main goal of this Master thesis was to compare two statistical models, based on the same clinical marker, one using a dichotomized outcome variable (logistic regression) and one using a continuous outcome (random coefficients model). In particular, we checked which of the two models is better suited for the estimation the probability of progression on diabetic patients with CKD.

The main results of this Master thesis are: First, the estimated probability of progression by the longitudinal model in the simulation was always closer to the true probability of progression than the probability of progression estimated by the logistic model. Moreover, the variability of the estimation of the probability of progression in the logistic models was higher than in the longitudinal models when using global Box-Cox parameters; for local Box-Cox parameters, the variability of the estimation was much higher in the longitudinal models. For equally spaced baseline UACR the variability of the progression probability estimation is always (for both global and local Box-Cox parameters) higher in the logistic than in the longitudinal models (see appendix).

Second, the power of the risk factor x in the longitudinal models was always higher than in the logistic models in the simulation. Furthermore, the power to detect an effect of x depends to a large degree on the sample size. Reducing (or increasing) the sample size by half (or by twice as much) led to an enormous drop (increase) in power (see Table 8).

Third, the misspecified LOG models behaved not too badly. Models using log UACR achieved comparable (but not equal) results and consequently similar interpretations as the models which used Box-Cox transformed log UACR (BCLOG models). This was the case in both the simulation study and the analysis of the ONTARGET-SysKid dataset. We noticed, that if the data are not properly transformed, then the probability of progression is underestimated, and in particular the difference between the risk groups x is underestimated in the simulation study. The reason is that by using not properly transformed data the residuals of the longitudinal model are not normally distributed. Therefore, the BCLOG model should always be preferred over the LOG model – if the true Box-Cox parameters were known. However, in practice the true transformation is not known and must be estimated from the data. A still unexplored approach to solving the underestimation of the probability of progression in the longitudinal LOG model for risk group x is to not consider a normal distribution, ϕ, but to use the empirical distribution of the residuals, $\hat{f}$, instead.

In the sequel, the logistic model refers to the logistic BCLOG model and the longitudinal model to the longitudinal BCLOG model.

Fourth, the probability of progression is highly overestimated by the logistic models in the simulation. Using local Box-Cox parameters instead of global ones for the transformation of log UACR re-

duces this effect slightly, i.e. without solving the problem of the overestimation (see Tables 24 and 25).

Fifth, the probability of progression is underestimated by the longitudinal models in the simulation. Surprisingly, using local Box-Cox parameters instead of global ones increases this underestimation bias (see Tables 24 and 25). One explanation for that could be that our method to search these Box-Cox parameters is not good enough. Our underlying longitudinal mixed model, on which we minimized the standard deviation of the difference between its standardized residuals and the quantiles of a standard normal distribution in order to find suitable Box-Cox parameters, included only time as fixed effect and the patient's individual intercepts and slope as random coefficients. Perhaps it would have been better not to use such a simple model, but to include other coefficients (such as sex, age, BMI, etc.) in the model as well to reduce the bias of the Box-Cox parameters estimation. In the two scenarios, in which the Box-Cox parameters were optimized on each simulated dataset, we additionally observed that the parameter c was highly underestimated.

Sixth, assessing model improvement in the logistic and longitudinal model showed in the simulation that the c-index and correlation of predicted probabilities of progression and observed progression status are slightly higher in the longitudinal model than in the logistic, while the IDI is slightly lower in the longitudinal model compared to the IDI of the logistic model. In summary, all measures of model improvement indicate that the risk factor seems to explain little in addition to the baseline UACR in the simulation. This is clearly revealed by plotting the predicted progression probabilities (see Figures 18 and 19). There is large variation in the estimated progression probabilities caused by the baseline UACR, but only little difference in the curves estimated for both groups of x. As the longitudinal model is closer to the data generation procedure in the simulation, the outcome is already better predicted by the baseline UACR and the risk factor x has even less additional value than in the logistic model, which is misspecified.

Seventh, assessing model improvement in the logistic and longitudinal model on the ONTARGET-SysKid dataset results in no mentionable differences between the logistic and longitudinal models. All measures of model improvement indicate that the randomization groups seem to explain little in addition to the baseline UACR. We only observe that the estimated probability of progression for the randomization group Combination is slightly lower than for the other two randomization groups in the analysis of the ONTARGET-SysKid study. This is the case for both models, i.e. the logistic and the longitudinal model, as the predicted progression probabilities show (see Figures 29 and 31).

One striking result was that the true probability of progression and the apparent progression rates of the patients were different. For example, in the simulation with global Box-Cox parameters the mean of the true probability of progression was 14 %, while the mean of the progression rate was

18 % for non-exposed subjects ($x = 0$). For exposed subjects ($x = 1$) this difference was even higher, e.g. the mean of the true probability of progression was 17 % and the mean of the progression rate was 25 %. Why is there such a difference between the true and apparent progression rates? The answer lies in the inclusion criterion of the ONTARGET-SysKid study, which incorporated only normo- or microalbuminuric patients, and in the albuminuria classification which was based on a single measurement. Measurement of albuminuria is prone to a considerable measurement error, if a single-spot urine sample is used (as in this study). If the measurement of albuminuria at baseline was by chance higher than the cut-off point for declaring macroalbuminuria (above 33.9 mg/mmol), then the patient was not included in the study. Such patients may, however, have lower values at subsequent measurements. Similarly, patients may have by chance a lower value at the first measurement, and show a 'progression' if their values are higher at subsequent measurements. While the first type of patients does not enter the study at all (they do not contribute to the denominator of the apparent progression rate), the latter will contribute to both numerator and denominator, and therefore a bias is induced in the estimation of the raw progression rate as described by Heinze and Neubauer (2012). Furthermore, any analysis that uses the binary progression indicator as outcome variable is biased, too. The longitudinal model can correct some of this bias as the random measurement error is a quantity that is considered and explicitly estimated by the model. Therefore, the estimated progression rates are lower in the longitudinal model, and, as our simulation revealed, are almost unbiased if compared to their theoretical values. One solution to minimize the difference between the true probability of progression and the progression rate would be to include patients, whose UACR measurements at baseline are even higher than the macroalbuminuria cut-off point (e.g., the cut-off point could be increased by twice the measurement error). Another solution would be to reduce the measurement error at baseline by measuring UACR not only once but e.g. three times and then taking the mean of these three measurements. This approach would decrease the measurement error by a rate of $\sqrt{3}$ (cf. Heinze and Neubauer 2012).

The unbiased estimation of progression rates in the longitudinal model comes at the cost of the assumption of linear trajectories of albuminuria at the transformed (i.e., the Box-Cox) scale. This assumption may or may not hold. Logically, the model optimizes the transformation given this assumption, but the danger is that by the Box-Cox transformation, irrelevant differences in very low albuminuria values are modeled, which can lead to a reduced power. A further disadvantage of the longitudinal model is that it cannot accommodate a combined endpoint (e.g., including doubling of serum creatinine or incidence of end stage renal disease). One way to analyze such combined endpoints are so-called joint models, which simultaneously model longitudinal trajectories and a time-to-event variable (cf. Rizopoulous 2010).

Is the optimization of the Box-Cox parameters on each simulated dataset reasonable or is it better to use fixed Box-Cox parameters for the transformations? On the one hand, using fixed Box-Cox parameters in the simulation saves computing time. But on the other hand, without the optimization of Box-Cox parameters on each dataset the assumption of normally distributed residuals may not be fulfilled in the longitudinal models. This may lead to missing important predictors and thus to an incorrect estimation of the power of the risk factor x in the longitudinal models of the simulation. Our study gives some rise to believe that it might be preferable to use a fixed transformation (which may lead to not perfectly normally distributed residuals) than to use the same data to estimate the transformation parameters, if only relatively few data are available (N = 1041 in the simulation). In the simulation we observed a severely increased variability of the progression probability estimates by optimizing the BC parameters in each simulated data sets. In the analysis of the ONTARGET-SysKid study, however, more data were available, which might stabilize the BC parameters. We conclude that in practical applications, the variability of the optimized BC parameters and its impact on the final estimates should be evaluated, e.g., using bootstrap.

One of the limitations of this Master thesis is that no more than three repeated measures of urine albumin excretion could be considered in the longitudinal models, since UACR was only measured at baseline and at the 2 and 5 years visits in the ONTARGET study. It would be interesting to find out what happens to the comparison of the logistic and longitudinal model when there are more than just three repeated UACR measurements. Therefore it might be appealing to further investigate the comparison of these two models on the BENEDICT study. BENEDICT is short for Bergamo Nephrologic Diabetes Complications Trial and it is a multicenter, double-blind, placebo-controlled, randomized study that examined the effects of the angiotensin-converting-enzyme inhibitor trandolapril in combination with the non-dihydropyridine calcium-channel blocker verapamil, trandolapril alone, verapamil alone and placebo on the incidence of microalbuminuria as described by Ruggenenti et al (2004). In this study urine albumin excretion was measured on a total of 1204 subjects at randomization and every six months thereafter for a median of 3.6 years (Ruggenenti et al 2004). The randomization groups were of equal size with approximately 300 subjects per group (Ruggenenti et al 2004).

Another limitation of this Master thesis is that the Box-Cox parameters were optimized only on each of the 1000 simulated datasets in two scenarios because of limited computing resources. For each scenario, the simulation with local Box-Cox parameters ran a bit more than two whole days (50 hours) while the simulation with global Box-Cox parameters took 'only' about 8 hours.

A further limitation of this Master thesis is that in the two scenarios, in which the Box-Cox parameters were optimized on each of the 1000 simulated datasets, new datasets were generated. Retrospectively, one could have used the same datasets to allow a direct comparison between the results of the simulation using global Box-Cox parameters to those of the simulation using local Box-Cox parameters in the log UACR transformations.

By means of conclusion, the longitudinal model on Box-Cox transformed log UACR should be the first choice for analysis because of its higher accuracy and unbiasedness. We have shown and exemplified how quantities describing binary outcomes that researchers are more familiar with (progression or no progression) can be derived and compared. R software for computing such derived quantities is contained in the appendix of this thesis.

6 List of references

Altman, D. G., Poyston, P. (2006) The cost of dichotomising continuous variables. British Medical Journal 332(7549): 1080

Anderson, J. J. (2007) Mean changes versus dichotomous definitions of improvement. Statistical Methods in Medical Research 16: 7-12

Boucquermont, J., Heinze, G., Jager, K. J., Oberbauer, R. and Leffondre, K. (2014) Regression methods for investigating risk factors of chronic kidney disease outcomes: the state of the art. BMC Nephrology 15: 45

Box, G.E. P. and Cox, D. R. (1964) An analysis of transformations. Journal of Royal Statistical Society *Series B (Methodological)* Vol. 26, No. 2: 211-252

Brown, H. and Prescott, R. (1999) Applied mixed models in medicine. New York, Wiley

Cheng, J., Edwards, L. J., Maldonado-Molina, M. M., Komro, K. A., Muller, K. E. (2009) Real longitudinal data analysis for real peple: Building a good enough mixed model. Statistics in Medicine 29: 504-520

Dunkler, D. , Dehghan, M., Teo, K. K., Heinze, G., Gao, P., Kohl, M., Clase, C. M., Mann, J. F. E., Yusuf, S.,Oberbauer, R. (2013) Diet and Kidney Disease in High-Risk Individuals with Type 2 Diabetes Mellitus. Journal of the American Medical Association. Internal Medicine 173(18): 1682-1692

Harrell, F. E. Jr.(2001) Regression Modeling Startegies - with Applications to Linear Models, Logistic Regression and Survival Analysis, Springer, 215-330

Heinze, G. and Neubauer, M. (2012) Application of R software to analyze CKD progression using longitudinal models for continuous outcomes, unpublished work package report from SysKid, work package 3

Mann, J. F. E., Schmieder R. E., Dyal L., Schumacher H., Pogue J., Wang X., Maggioni A., Budaj A., Chaithiraphan S., Dickstein K., Keltai M., Metsärinne K., Oto A., Parkhomenko A., Piegas, L. S., Svendsen T. L., Teo, K. K. and Yusuf S. (2008) Renal outcomes with telmisartan, ramipril, or both, in people at high vascular risk (the ONTARGET study). Lancet 372: 547-53

National Kidney Disease Education Program:
http://nkdep.nih.gov/resources/quick-reference-uacr-gfr.shtml (accessed on 10 July 2013)

National Kidney Foundation: KDOQI Clinical Practice Guidelines for Chronic Kidney Disease: http://www.kidney.org/professionals/kdoqi/guidelines_ckd/p4_class_g1.htm (accessed on 22 April 2014) or http://www.kidney.org/kidneydisease/aboutckd.cfm (accessed on 18 July 2014)

Peacock, J. L., Sauzet, O., Ewings, S. M. and Kerry, S. M. (2012) Dichotomising continuous data while retaining statistical power using a distributional approach. Statistics in Medicine 31: 3089-3103

Pencina, M. J., D'Agostino Sr, R. B., D'Agostino Jr, R. B. and Vasan, R. S. (2008) Evaluating the added predictive ability of a new marker: From area under the ROC curve to reclassification and beyond. Statistics in Medicine 27: 157-172

R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/

Rizopoulos, D. (2010) JM: An R Package for the Joint Modelling of Longitudinal and Time-to-Event Data. Journal of Statistical Software. Volume 35. Issue 9

Ruggenenti, P., Fassi, A., Ilieva, A. P., Bruno, S., Iliev, I. P., Brusegan, V., Rubis, N., Gherardi, G., Arnoldi, F., Ganeva, M., Ene-Iordache, B., Gaspari, F., Perna, A., Bossi, A., Trevisan, R., Dodesini, A. R., Remuzzi, G. (2004) Preventing Microalbuminuria in Type 2 Diabetes. The New England Journal of Medicine 351: 1941-1951

SAS (2014): Mixed models theory, http://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#statug_mixed_sect022.htm (accessed on 3 September 2014)

Steyerberg, S. W. (2009). Clinical Prediction Models, Springer, New York

SysKid: www.syskid.eu (accessed on 10 July 2013)

Fedorov, V., Mannino, F., Zhang, R. (2009) Consequences of dichotomization. Pharmaceutical Statistics 8: 50-61

Vittinghoff, E., Glidden D. V., Shiboski, S. C., McCulloch, C. E. (2005) Regression Methods in Biostatistics: Linear, Logistic, Survival, and Repeated Measures Models. Springer Verlag: 157-211

7 Appendices

The appendices are composed of four parts: the explanations of all self-written R functions (subsection 7.1), all probability of progression plots (subsection 7.2), the names of all supplementary files (subsection 7.3) and my curriculum vitae (subsection 7.4).

7.1 R functions

In this subsection we present a description for all self-written R functions which were used in the simulation process (subsection 7.1.1) and in the analysis of the ONTARGET-SysKid dataset (subsection 7.1.2). In order to easily gain an insight into what these R functions do, a description for every function is provided in the R-help style (with the subsections Description, (Requirements), Usage, Arguments, Details, Value and Examples).

7.1.1 R functions for simulation

This subsection contains a description of all self-written R functions which were used in the simulation. These functions are: `boxcox`, `boxcox2`, `objective`, `inboxcox`, `cutpoint`, `progression`, `simdata`, `progr.lme`, `prob.progr.true`, `simProgaPower`, `simTable`, `simPlots`, `simProgaPoweropt`, `simTableopt`, `simPlotsopt`, `compute.c` and `compute.idi`.

7.1.1.1 Boxcox function

Description

This function transforms log UACR into BC log UACR in order to obtain standard normally distributed residuals when applying a mixed model.

Usage

```
boxcox(x, lambda= 0.5011719, c=2.8494141, mue=1.494655, sigma=0.8691764)
boxcox2(par, x, mue=0, sigma=1)
boxcox3(lambda=0.67, x, mue=1.041583, sigma=1.17939)
boxcox4(lambda=0.5, x)
```

Arguments

`par`	vector containing λ and c
`lambda`	parameter λ
`c`	parameter c
`x`	log UACR which will be transformed; UACR in mg/mmol
`mue`	mean of Box-Cox transformed values (μ)
`sigma`	standard deviation of Box-Cox transformed values (σ)

Details

λ and c are the results of the optimization process and μ and σ are the results of the `objective` function (see next R function) when plugging in the optimal λ and `c`.

In this Master thesis four different R functions for Box-Cox transformations were written. The function `boxcox2` is only used in combination with R's built-in function `optim` and the self-written function `objective` in the simulation. If the Box-Cox parameters are not optimized, the `boxcox` function is chosen. This function uses by default the global Box-Cox parameters. The Box-Cox transformed value E_1 in the functions `boxcox` and `boxcox2` is given by

$$E_1 = \begin{cases} \dfrac{(\log(U) + c)^{\lambda} - 1}{\lambda}, & \lambda \neq 0 \\ \log(\log(U) + c), & \lambda = 0 \end{cases}$$

The function `boxcox4` is only used in combination with R's built-in function `optimize` and the self-written function `target` in the analysis of the ONTARGET-SysKid datset. If the Box-Cox parameter λ is not optimized, then the `boxcox3` function is chosen. This function uses by default the global Box-Cox parameter. The Box-Cox transformed value E_2 in the functions `boxcox3` and `boxcox4` is given by

$$E_2 = \begin{cases} \dfrac{(\log(U))^{\lambda} - 1}{\lambda}, & \lambda \neq 0 \\ \log(\log(U)), & \lambda = 0 \end{cases}$$

Note that the function `boxcox3` generates NAs when inserting negative values into `x`.

Value

Box-Cox transformed log urine albumin excretion values

Examples

```
# example for boxcox
# load function invboxcox
loguacr<-invboxcox(rnorm(100,mean=0.315,sd=1.766))
boxcox(loguacr)

# example for boxcox2
library(mnormt) # required for simdata
library(nlme) # required for objective
# load the functions boxcox, invboxcox, cutpoint, progression and simdata
set.seed(349058)
testdata<-simdata(n=1041,gamma11=0.05)
# load boxcox2 and objective function
# optimize the Box-Cox parameters:
result<-optim(par=c(1,2),fn=objective, ret=F, x=testdata$longi$logalb,
```

```
time=testdata$longi[,"time"], ID=testdata$longi[,"id"])
result # This operation can take a few minutes.

# example for boxcox3
# load functions invboxcox and winsor
set.seed(3490583)
loguacr<-invboxcox(rnorm(100,mean=0.315,sd=1.766))
trans<-winsor(loguacr)
boxcox3(x=trans)

# example for boxcox4
library(mnormt)
library(nlme)
# load the functions boxcox, invboxcox, cutpoint, progression and simdata
set.seed(3458)
testdata<-simdata(n=1041,gamma11=0.05)
dat<-winsor(testdata$longi[,"logalb"])
# load boxcox4 and target function
# optimize the Box-Cox parameter lambda
result<-optimize(f=target,interval=c(-2,5),ret=F, x=dat,
time=testdata$longi[,"time"], ID=testdata$longi[,"id"], maximum=FALSE)
result # This operation can take a few seconds.
```

7.1.1.2 Objective function

Description

The `objective` function for optimization of the Box-Cox parameters in the simulation was defined as the standard deviation of the difference of the quantiles of a standard normal distribution and the standardized residuals of a random coefficients model which uses the standardized Box-Cox transformed log urine albumin excretion values as dependent variable.

Requirements

`boxcox2` function

Usage

```
objective(par,x,time, ID, ret=T, plot=F)
```

Arguments

`par`	vector consisting of λ and c
`x`	log urine albumin excretion values; UACR in mg/mmol
`time`	time of measurements in years
`ID`	identification of patients
`ret`	defines what will be returned
`plot`	spezifies if a qqplot is returned or not

Details

By default ret is set to TRUE. This means that the standard deviation of the difference of the quantiles of a standard normal distribution and the standardized residuals of a random coefficients model which uses the Box-Cox transformed log UACR values as dependent variable as well as the mean and the standard deviation of the Box-Cox transformed data are returned. If ret=FALSE, then the standard deviation of the difference of the quantiles of a standard normal distribution and the standardized residuals of a random coefficients model is returned, but not the mean and standard deviation of the Box-Cox transformed data.

The parameter plot is by default FALSE to suppress the qqplots.

Value

This function returns the standard deviation of the difference of the quantiles of a standard normal distribution and the standardized residuals of a random coefficients model which uses the Box-Cox transformed log UACR values as dependent variable. If ret = TRUE the mean and the standard deviation of the Box-Cox transformed log UACR are also returned.

Example

```
library(mnormt); library(nlme)
# load the functions boxcox, invboxcox, cutpoint, progression and simdata
set.seed(349058)
testdata<-simdata(n=1041,gamma11=0.05)
# load boxcox2 function
objective(c(1,2), x=testdata$longi$logalb, time=testdata$longi$time,
ID=testdata$longi$id, showplot=TRUE,ret=TRUE)

# optimize the Box-Cox parameters:
result<-optim(par=c(1,2),fn=objective, ret=F, x=testdata$longi$logalb,
time=testdata$longi[,"time"], ID=testdata$longi[,"id"])
result# This operation may take some minutes.
```

7.1.1.3 Inboxcox function

Description

The `invboxcox` function is the inverse function to the `boxcox` function. Therefore this function uses by default the same parameters as the `boxcox` function.

Usage

```
invboxcox(x, lambda= 0.5011719, c=2.8494141, mue=1.494655, sigma=0.8691764)
```

Arguments

x	Box-Cox transformed log UACR
lambda	parameter λ
c	parameter c
mue	mean of Box-Cox transformed log UACR
sigma	standard deviation of Box-Cox transformed log UACR

Details

Please note that when x is below -4.0152, this function returns NA.

Value

Log urine albumin excretion values

Example

```
library(mnormt)
invboxcox(rnorm(10, mean=0.315, sd=1.766))
```

7.1.1.4 Cut-off point function

Description

The cutpoint function calculates the cut-off point on the log UACR scale for individual patients depending on their baseline values, which can then be used to define progression.

Usage

```
cutpoint(logbase)
```

Arguments

logbase	log (urine albumin excretion measured in mg/mmol) at baseline

Value

This function returns the cut-off point used for defining progression in CKD of a patient.

Example

```
# load invboxcox
set.seed(1324345349)
loguacr<-invboxcox(rnorm(10,mean=0.315,sd=1.766))
# patients with macroalbuminuria are excluded:
logbase<-loguacr[loguacr<log(33.9)]
cutpoint(logbase)
```

7.1.1.5 Progression function

Description

The `progression` function checks whether CKD progressed in a patient or not.

Requirements

`cutpoint` function

Usage

```
progression(logbase, logalb5)
```

Arguments

`logbase`	log urine albumin excretion value at baseline; UACR is measured in mg/mmol
`logalb5`	log urine albumin excretion value at 5 years; UACR is measured in mg/mmol

Value

This function returns a number which is either zero or one. Zero means the patient did not experience a progression in CKD after five years and one denotes that the patient is progressing in CKD.

Example

```
# load invboxcox
library(mnormt)
set.seed(929049)
logub<- invboxcox(rnorm(10, mean=0.315, sd=1.766))
logu5<- invboxcox(rnorm(10, mean=0.315, sd=1.766))
progression(logbase=logub, logalb5=logu5)
```

7.1.1.6 Data Generation Function

Description

The `simdata` function generates new datasets of fictive patients for both, the logistic and the longitudinal model.

Requirements

It is required to load the `boxcox`, `invboxcox`, `cutpoint` and `progression` function before using this function. Furthermore, the R-package `mnormt` is required.

Usage

```
simdata(n=1041, gamma00=0, gamma01=0, gamma10=0, gamma11=0.05,
sigma0 = rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)),
sigma1 = rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)),
RSQE = 0.6362203, meanx=0.5)
```

Arguments

n	number of patients
gamma00	average y at baseline in group $x = 0$
gamma01	increase in y per year in group $x = 0$
gamma10	average y at baseline in group $x = 1$
gamma11	increase in y per year in group $x = 1$
sigma0	covariance matrix of random intercept and slope in group $x = 0$
sigma1	covariance matrix of random intercept and slope in group $x = 1$
RSQE	square root of the residual variance (independent of individual or time point)
meanx	expected value of binary risk factor x

Value

This function returns a list which consists of two data frames (`longi` and `logist`) and the vector `nbase`. The data frame `logist` consists of the following variables for each patient: `id`, `x`, `progr`, `covar` and `nbase`. Patient identification is `id`; `x` specifies whether the patient has a certain risk factor (`x = 1`) or not (`x = 0`); `progr` denotes if the patient developed progression in CKD (`progr = 1`) or not (`progr = 0`); `covar` is the difference between the cut point and the baseline UACR and `nbase` is UACR at baseline (measured in mg/mmol).

The contents of the data frame `longi` is: `id`, `x`, `time`, `nalb` and `logalb`. The first two variables, `id` and `x`, are the same as in the data frame `logist`, `time` specifies the time point of the UACR measurements (in years), `nalb` are simulated Box-Cox log UACR and `logalb` are the `invboxcox` transformed `nalb`.

The vector `nbase` contains the baseline Box-Cox log UACR of the simulated patients.

Example

```
# load boxcox, invboxcox, cutpoint and progression function
library(mnormt)
simdata(n= 10, gamma11=0.05)
```

7.1.1.7 Probability of progression function

Description

This function calculates the probability of progression estimated by a longitudinal model.

Requirements

R-package `nlme` and self-written R-function `boxcox`

Usage

```
progr.lme(fit=fitl, model="u1", X=T, x=1, time=5, logbase=-0.1127955,
cutpoint=NULL,bcpar=c(0.5011719,2.8494141), bcmu=1.494655, bcsig=0.8691764)

progr.lme2(fit=fitl, model="u1",X=T, x1=1, x2=0, time=5,
logbase= -0.1127955, cutpoin=NULL)
```

Arguments

`fitl`	fit object from a longitudinal model (`lme`)
`model`	defines the type of data hidden in the fit object: `u1` used normally distributed UACR (BCLOG model) and `u2` used log urine albumin excretion (LOG model) as input
`x`	defines to which risk group the patient belongs to
`x1`	specifies if the patient belongs to Ramipril; it is either 0 (i.e. no) or 1 (i.e. yes)
`x2`	specifies if the patient belongs to Combination; it is either 0 (i.e. yes) or 1 (i.e. no)
`X`	if X=T, then `x` or `x1` and `x2` is/are used in the models as a co-variable.
`time`	time point in years for which one seeks the probability of progression
`logbase`	log urine albumin excretion at baseline
`cutpoint`, `cutpoin`	if no cut-off point is specified, cut points are calculated using the `cutpoint` function.
`bcpar`	local Box-Cox parameters (result from optimization)
`bcmu`	mean for the Box-Cox transformation (result from optimization)
`bcsig`	standard deviation for the Box-Cox transformation (result from optimization)

Details

By default the median of the logarithmized urine albumin excretion of a random subset of the ON-TARGET-SysKid dataset (n=200) is used in `logbase`.

If `bcpar`, `bcmu` and `bcsig` are not specified, the global Box-Cox parameters are used.

The difference between `progr.lme` and `progr.lme2` is that in `progr.lme` the patients are divided into two groups (`x` $\in$ {0, 1}) whereas in `progr.lme2` the patients are divided into three groups (by the dummy variables: `x1` and `x2`). If `x1 = 1`, then the patient belongs to Ramipril and if `x2 = 1`, then the patient belongs to Combination. If both `x1` and `x2` are both zero then the probability of progression is calculated for a patient who belongs to Telmisartan.

Value

This function returns a list which consists of the estimate of log UACR, standard error, unadjusted estimate, standard error of the unadjusted estimate, G-matrix, G* (the adjusted covariance matrix of random effects given known baseline value) and the probability of progression.

Example

```
# example for progr.lme:
# load functions: boxcox, invboxcox, cutpoint, progression and simdata
library(nlme)
library(mnormt)
set.seed(345998)
daten<-simdata(n= 10, gamma11=0.05)
fit1<-lme(data=daten$longi, fixed=logalb~time+x+time:x, random=~1+time|id)
logb<- seq(log(0.01),log(33.8),by=(log(33.8)-log(0.01))/99)
prob<-numeric(length(logb))
for(i in 1:length(logb))
{prob[i]<-progr.lme(fit=fit1, model="u2",logbase=logb[i])$prob.progr}
plot(logb, prob,type="l",xaxt="n",xlab="UACR at baseline in mg/mmol")
axis(side=1,at=seq(min(logb),max(logb),length=10),labels= round(exp(seq(min(logb),
 max(logb), length=10)), digits=1))

# example for progr.lme2:
# load functions: boxcox, invboxcox, cutpoint, progression and simdata
library(nlme)
library(mnormt)
set.seed(3045034)
longi<-matrix(0,150,2)
colnames(longi)<-c("x1","x2")
longi<-as.data.frame(longi)
x1<-sample(c(0,1),50,replace=TRUE, prob=c(0.7,0.3))
x2<-numeric(50)
for(i in 1:length(x1)) if(x1[i]!=1) x2[i]<-sample(c(0,1),1)
longi$x1<-rep(x1,each=3)
longi$x2<-rep(x2,each=3)
daten<-simdata(n= 50, gamma11=0.05)
longi<-cbind(daten$longi,longi)
fit1<-lme(data=longi, fixed=logalb~time+x1+time:x1+x2+time:x2, random=~1+time|id)
logb<- seq(log(0.01),log(33.8),by=(log(33.8)-log(0.01))/99)
prob<-numeric(length(logb))
for(i in 1:length(logb))
{
 prob[i]<-progr.lme2(fit=fit1, x1=longi$x1[i],x2=longi$x2[i], model="u2",
 logbase=logb[i])$prob.progr
}
```

```
par(mfrow=c(1,3))
plot(logb[longi$x1==1],prob[longi$x1==1],type="l", main="group 1", xaxt="n",
 ylab="probability of progression", xlab="UACR at baseline in mg/mmol")
axis(side=1,at=seq(min(logb),max(logb),length=10), labels= round(exp(seq(min(logb),
 max(logb),length=10)), digits=1))
plot(logb[longi$x2==1], prob[longi$x2==1],type="l", main="group 2",
 ylab= "probability of progression", xaxt="n", xlab="UACR at baseline in mg/mmol")
axis(side=1,at=seq(min(logb),max(logb),length=10),labels=
 round(exp(seq(min(logb),max(logb),length=10)),digits=1))
plot(logb[longi$x1==0 & longi$x2==0],prob[longi$x1==0 & longi$x2==0], type="l",
 ylab= "probability of progression",main="group 3", xaxt="n",
 xlab="UACR at baseline in mg/mmol")
axis(side=1,at=seq(min(logb),max(logb),length=10),
 labels= round(exp(seq(min(logb),max(logb),length=10)),digits=1))
par(mfrow=c(1,1))
```

7.1.1.8 True probability of progression function

Description

To find the true value for the probability of progression, the function `prob.progr.true` was written and tested for different γ_{11}.

Requirements

`boxcox` function

Usage

```
prob.progr.true(x=1, time=5, logbase=-0.1127955, gamma00=0, gamma01=0,
gamma10 = 0, gamma11 = 0.2, RSQE = 0.6362203,
G = rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)))
```

Arguments

`x`	defines the risk group
`time`	time point for which one seeks the probability of progression
`logbase`	log urine albumin excretion at baseline; UACR in mg/mmol
`gamma00, gamma01, gamma10, gamma11`	see `simdata` function
`G`	variance-covariance matrix of the random effects
`RSQE`	see `simdata` function

Details

By default the median of the logarithmized urine albumin excretion of a random subset of the ONTARGET-SysKid dataset (n=200) is used in `logbase`. And, by default the variance-covariance matrix of the random effects (`G`) is taken from the model `lme(albBC~time,random=~1+time|ID,data=longi)` on the ONTARGET-SysKid dataset (see G-matrix in subsection 4.2.1.1).

Value

This function returns a list which consists of the estimate of log UACR, standard error, unadjusted estimate, standard error of the unadjusted estimate, G-matrix, G* (the adjusted covariance matrix of random effects given known baseline value) and the true probability of progression.

Example

```
logb<- seq(log(0.01), log(33.8), by=(log(33.8)-log(0.01))/99)
trueprob<-numeric(length(logb))
for(i in 1:length(logb))
{
 trueprob[i]<-prob.progr.true(x=0, time=5, logbase=logb[i],
 gamma11=0.05)$prob.progr
}
plot(logb, trueprob, type="l", xaxt="n", xlab="UACR at baseline in mg/mmol")
axis(side=1, at=seq(min(logb),max(logb),length=10),
 labels= round(exp(seq(min(logb),max(logb),length=10)), digits=1))
```

7.1.1.9 Simulation function - no optimization of BC parameters

In order to minimize the simulation time, the simulation function is split into three parts. In the first part, datasets are simulated and the fit objects calculated and saved. In the second part, the results of the simulated patients which are gained from the fit objects, such as for example the estimated probability of progression, are calculated and saved. And in the third part, the probabilities of progression are calculated for 100 equally-spaced baseline values ranging from 0.01 to 33.79 mg/mmol. This last part is executed to generate nice plots (see subsection 7.2).

7.1.1.9.1 Simulation function part 1

Description

The aim of this function is to simulate urine albumin excretion levels from a plausible distribution, to fit the logistic regression and the random coefficients model and to investigate the power of the logistic regression and the longitudinal models.

Requirements

When using the following function, it is important to run all the functions from the file functions.r first, as these functions are used in the simulation function. Furthermore, this function requires the R-packages `mnormt` and `nlme`.

Usage

```
simProgaPower(n=1041, nsim=100, gamma00=0,gamma01=0,gamma10=0,gamma11=0.05,
sigma0=rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)),
sigma1=rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)), RSQE=0.6362203,
meanx=0.5)
```

Arguments

`n`	number of patients
`nsim`	number of simulations
`gamma00`	average y at baseline in group $x = 0$
`gamma01`	increase in y per year in group $x = 0$
`gamma10`	average y at baseline in group $x = 1$
`gamma11`	increase in y per year in group $x = 1$
`sigma0`	covariance matrix of random intercept and slope in group $x = 0$
`sigma1`	covariance matrix of random intercept and slope in group $x = 1$
`RSQE`	square root of the residual variance (independent of individual or time point)
`meanx`	expected value of binary risk factor

Details

To roughly explain this function, first, the group membership `x` is randomly drawn from a binominal distribution with a probability of success on each trial of 50 % (`meanx` $= 0.5$). Then, Box-Cox transformed log urine albumin excretion values are generated and saved. In order to get log UACR the Box-Cox log UACR values are transformed using the `invboxcox` function with the global Box-Cox parameters. After that, the logistic and the longitudinal models are fitted. Both methods are applied four times: with and without the marker `x` and using either the log UACR or BC log UACR values. In the end, progression rates, apparent relative risks and the power to detect an association with the risk factor `x` in the two models are calculated.

Whenever a longitudinal model did not converge, the model from the previous simulation was used for further calculations. This occurred in not more than 0.1 % of all cases. "In most cases, when a

mixed model algorithm does not converge, it is typically assumed to be due to difficulties with the estimation of the covariance parameters" (Cheng, 2010: 512).

Value

This function returns a list which includes the power of the logistic model (`power_logistic`), the power of the longitudinal model (`power_longitudinal`), the mean and standard deviation of the progression rates in group `x = 0` and group `x = 1` (`m.prograte.x0`, `sd.prograte.x0` and `m.prograte.x1` and `sd.prograte.x1`, respectively), the apparent relative risk of progression for the patients (`mean.relative.risk`), which is defined as the ratio of the progression rates for these two groups, the coefficients of the longitudinal models (`coef.longi.u2`, `coef.longi.u1`), the simulated datasets (`simdata`), the baseline UACR values of all patients in each dataset (`nbase`), all fit objects (`fitb.u2`, `fitbx.u2`, `fitb.u1`, `fitbx.u1`, `fitl.u2`, `fitlx.u2`, `fitl.u1` and `fitlx.u1`) and the number how often the longitudinal model did not converge. The variables `count.u2`, `count.u2x`, `count.u1` and `count.u1x` contain these numbers for each longitudinal model.

Example

```
# load functions: boxcox, invboxcox, cutpoint, progression, simdata, prog.lme
# and prob.progr.true
library(nlme)
library(mnormt)
set.seed(3294)
simpart1<-simProgaPower(n=100, nsim=20, gamma11=0.05)# takes approx. 50 seconds
```

7.1.1.9.2 Simulation function part 2

Description

In this part, the probability of progression estimated by each model, as well as the true probability of progression is calculated. Furthermore, the expected relative risk and the expected odds ratio regarding the risk of progression are determined.

Requirements

Again, all the functions from the file functions.r have to be run first. Moreover, it is required to load the results of the part 1 of the simulation (applying the self-written R function `simProgaPower`).

Usage

```
simTable(gamma11=0, simbeta)
```

Arguments

`gamma11` increase in y per year in group $x = 1$

`simbeta` R-object, that includes all results from part 1 of the simulation

Details

See subsection 4.2.2 for a detailed definition of the expected relative risk and expected odds ratio regarding the risk of progression.

The argument `simbeta` is a list, that contains the following objects in this order: `power_logistic`, `power_longitudinal`, `m.prograte.x0`, `sd.prograte.x0`, `m.prograte.x1`, `sd.prograte.x1`, `mean.relative.risk`, `coef.longi.u2`, `coef.longi.u1`, `simdata`, `nbase`, `fitb.u2`, `fitbx.u2`, `fitb.u1`, `fitbx.u1`, `fitl.u2`, `fitlx.u2`, `fitl.u1`, `fitlx.u1`, `count.u2`, `count.u2x`, `count.u1` and `count.u1x`. A description of these objects is given in the previous function description.

Value

The function `simTable` returns a list consisting of a summary statistic of probability of progression from each model (`s.u1.fitb0`, `s.u2.fitb0`, `s.u1.fitb1`, `s.u2.fitb1`, `s.u1.fitl0`, `s.u2.fitl0`, `s.u1.fitl1` and `s.u2.fitl1`), as well as the true probability of progression for both groups of `x` as well as the summary statistics thereof (`trueprob.x0`, `trueprob.x1`, `s.true.0` and `s.true.1`), the expected relative risk (`rel.risk`) and the expected odds ratio (`OR`).

Example

```
# load functions: boxcox, invboxcox, cutpoint, progression, simdata, prog.lme
# and prob.progr.true
library(nlme)
library(mnormt)
set.seed(3294)
simpart1<-simProgaPower(n=100,nsim=20,gamma11=0.05)
simpart2<-simTable(gamma11=0.05,simbeta=simpart1)# takes approx. 32 seconds
```

7.1.1.9.3 Simulation function part 3

Description

In this part, the fit objects from the first part are loaded and the probability of progression from each model and the true probability of progression are calculated regarding 100 different baseline UACR ranging from 0.01 to 33.79 mg/mmol.

Requirements

Again, all the functions from the file functions.r have to be run first. Additionally, it is also required to load the results from part 1 of the simulation (function `simProgaPower`).

Usage

```
simPlots(gamma11=0, simbeta)
```

Arguments

`gamma11`	increase in y per year in group $x = 1$
`simbeta`	R-object, that includes all results from part 1 of the simulation

Details

A detailed definition of the expected relative risk and expected odds ratio regarding the risk of progression is provided in subsection 4.2.2.

The argument `simbeta` is a list, that contains the following objects in this order: `power_logistic`, `power_longitudinal`, `m.prograte.x0`, `sd.prograte.x0`, `m.prograte.x1`, `sd.prograte.x1`, `mean.relative.risk`, `coef.longi.u2`, `coef.longi.u1`, `simdata`, `nbase`, `fitb.u2`, `fitbx.u2`, `fitb.u1`, `fitbx.u1`, `fitl.u2`, `fitlx.u2`, `fitl.u1`, `fitlx.u1`, `count.u2`, `count.u2x`, `count.u1` and `count.u1x`. A description of all these objects is given in the function description of `simProgaPower`.

Value

The probability of progression estimated by each model (`BB.u2.prob.fitb.x0`, `BB.u2.prob.fitb.x1`, `BB.u1.prob.fitb.x0`, `BB.u1.prob.fitb.x1`, `BB.u2.prob.fitl.x0`, `BB.u2.prob.fitl.x1`, `BB.u1.prob.fitl.x0` and `BB.u1.prob.fitl.x1`) and the true probability of progression for each group of `x` (`BB.tureprob.x1` and `BB.trueprob.x0`) are calculated regarding 100 equally spaced baseline UACR ranging from 0.01 to 33.79 mg/mmol and are returned in a list.

Example

```
# load functions: boxcox, invboxcox, cutpoint, progression, simdata, prog.lme
# and prob.progr.true
library(nlme)
library(mnormt)
set.seed(3294)
simpart1<-simProgaPower(n=100,nsim=20,gamma11=0.05)
simpart3<-simPlots(gamma11=0.05,simbeta=simpart1)# takes approx. 31 seconds
```

7.1.1.10 Simulation function - optimization of BC parameters

Again, the simulation is split into three parts. The only difference to the previous simulation function is that for the BCLOG model the local Box-Cox parameters are used, i.e. the Box-Cox parameters are optimized on each simulated dataset.

7.1.1.10.1 Simulation part 1

Description

This function simulates datasets, searches the local Box-Cox parameters through optimizing the residuals of a longitudinal model with the purpose of meeting the normal distribution assumption, fits logistic and longitudinal models, calculates the progression rates of both groups of `x` and the power of the association of the risk factor `x` with progression of the models.

Requirements

All the functions from the file functions_opt.r have to be run first.

Usage

```
simProgaPoweropt(n=1041, nsim=100, gamma00=0,gamma01=0,gamma10=0,
gamma11=0.05, sigma0 = rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)),
sigma1= rbind(c(0.57108,0.0034829), c(0.0034829,0.0081404)),
RSQE=0.6362203, meanx=0.5)
```

Arguments

`n`	number of patients
`nsim`	number of simulations
`gamma00`	average y at baseline in group $x = 0$
`gamma01`	increase in y per year in group $x = 0$
`gamma10`	average y at baseline in group $x = 1$
`gamma11`	increase in y per year in group $x = 1$
`sigma0`	covariance matrix of random intercept and slope in group $x = 0$
`sigma1`	covariance matrix of random intercept and slope in group $x = 1$
`RSQE`	square root of the residual variance (independent of individual or time point)
`meanx`	expected value of binary risk factor

Details

First, BC log UACR values are generated; additionally they were transformed using the `invboxcox` function in order to get log UACR (for LOG models). On these log UACR values, the local Box-Cox

parameters are searched through optimization with R's built-in function `optim`. In a further step, the local parameters were inserted into `boxcox` function.

Whenever a longitudinal model did not converge, the model from the previous simulation was used for further calculations. This occurred in not more than 0.2 % of all cases.

Whenever a longitudinal model did not converge inside the objective function, which means that the local Box-Cox parameters could not be calculated for the simulated dataset, the global Box-Cox parameters were used instead. This occurred in not more than 0.6 % of all cases.

Value

This function returns a list which includes the power of the logistic model (`power_logistic`), the power of the longitudinal model (`power_longitudinal`), the mean and standard deviation of the progression rates in group `x = 0` and group `x = 1` (`m.prograte.x0`, `sd.prograte.x0` and `m.prograte.x1` and `sd.prograte.x1`, respectively), the apparent relative risk of the patients (`mean.relative.risk`) which is defined as the ratio of the progression rates for these two groups, the coefficients of the longitudinal models (`coef.longi.u2` and `coef.longi.u1`), the simulated datasets (`simdata`), the baseline Box-Cox log UACR for each dataset (`nbase`), all fit objects (`fitb.u2`, `fitbx.u2`, `fitb.u1`, `fitbx.u1`, `fitl.u2`, `fitlx.u2`, `fitl.u1` and `fitlx.u1`) and the number how often the longitudinal model did not converge (`count.u2`, `count.u2x`, `count.u1` and `count.u1x`). If the longitudinal model did not converge, then the mixed models from the previous iteration were used for the calculations.

In addition the local Box-Cox parameters are returned for each simulation step (`bcPAR1`, `bcPAR2`, `bcmu`, `bcsig` and `sdOPT`). And it was counted how often the global Box-Cox parameters were used instead of the local (`count.p`).

Example

```
# load functions: boxcox, boxcox2, objective, invboxcox, cutpoint,
# progression, simdata, prog.lme and prob.progr.true
library(nlme); library(mnormt)
set.seed(49303)
simopt1<-simProgaPoweropt(n=100,nsim=20,gamma11=0.05)# approx. 15 minutes
```

7.1.1.10.2 Simulation part 2

Description

Again, the only difference to the function `simTable` is that the local Box-Cox parameters instead of the global ones are used in this function for the BCLOG models.

Requirements

Again, all self-written R functions from the file functions_opt.r have to be run first.

Usage

```
simTableopt(gamma11=0, path, simopt)
```

Arguments

`gamma11`	increase in y per year in group $x = 1$
`path`	path to locate were the list of part 1 was saved
`simopt`	list from part 1 of the simulation

Value

This function returns a list which contains summary statistics of probabilities of progression estimated by each model (`s.u1.fitb0`, `s.u2.fitb0`, `s.u1.fitb1`, `s.u2.fitb1`, `s.u1.fitl0`, `s.u2.fitl0`, `s.u1.fitl1` and `s.u2.fitl1`), the true probabilities of progression for each group of `x` as well as the summary statistics thereof (`trueprob.x0`, `trueprob.x1`, `s.ture.0` and `s.ture.1`), the expected relative risk (`rel.risk`) and the expected odds ratio (`OR`) regarding the risk of progression.

Example

```
# load functions: boxcox, boxcox2, objective, invboxcox, cutpoint, progression,
# simdata, prog.lme and prob.progr.true
library(nlme)
library(mnormt)
set.seed(49303)
simopt1<-simProgaPoweropt(n=100,nsim=20,gamma11=0.05)
simopt2<-simTableopt(gamma11=0.05,simopt=simopt1) # takes approx. 33 seconds
```

7.1.1.10.3 Simulation part 3

Description

This function is similar to the function `simPlots`, the only difference is that the local Box-Cox parameters instead of the global ones are used in this function for the BCLOG models.

Requirements

Again, all self-written R functions from the file functions_opt.r have to be run first.

Usage

```
simPlotsopt(gamma11=0, path, simopt)
```

Arguments

`gamma11`	increase in y per year in group $x = 1$
`path`	path to locate were the list of part 1 was saved
`simopt`	list from part 1 of the simulation

Details

Either `path` or `simopt` have to be specified.

Value

This function returns a list which consists of the probability of progression estimated by each model (`BB.u2.prob.fitb.x0`, `BB.u2.prob.fitb.x1`, `BB.u1.prob.fitb.x0`, `BB.u1.prob.fitb.x1`, `BB.u2.prob.fitl.x0`, `BB.u2.prob.fitl.x1`, `BB.u1.prob.fitl.x0` and `BB.u1.prob.fitl.x1`) and the true probability of progression for each group of `x` (`BB.trueprob.x1` and `BB.tureprob.x0`) are determined on 100 equally spaced baseline UACR, ranging from 0.01 to 33.79 mg/mmol.

Example

```
# load functions: boxcox, boxcox2, objective, invboxcox, cutpoint, progression,
# simdata, prog.lme and prob.progr.true
library(nlme); library(mnormt)
set.seed(49303)
simopt1<-simProgaPoweropt(n=100,nsim=20,gamma11=0.05)
simopt3<-simPlotsopt(gamma11=0.05,simopt=simopt1)# takes approx. 33 seconds
```

7.1.1.11 C-Statistic Function

Description

This function calculates the c-statistics and the correlation of predicted probabilities of progression and observed progression status for the logistic and longitudinal models.

Requirements

The functions `compute.c` and `compute.c2` require each a fit object of the logistic model and one of the longitudinal model, as well as a dataset. It is also essential to specify which model type was used to calculate the fit objects (either `u1` for BCLOG models or `u2` for LOG models).
Additionally, the self-written R functions `invboxcox` (see subsection 7.1.1.3)and `progr.lme` (see subsection 7.1.1.7) are required for these functions.

Usage

```
compute.c(fitl, fitb, newdata, modeltype = NULL)
compute.c2(fitl, fitb, n, progr, x1, x2, covar, modeltype = NULL)
```

Arguments

`fitl`	fit object from a longitudinal model
`fitb`	fit object from a logistic model
`newdata`	test sample
`n`	number of patients in the dataset
`progr`	specifies whether a patient developed progression (`progr=1`) or not (`progr=0`)
`x1`	group membership: `x1=1` if patient belongs to Ramipril; `x1=0` else
`x2`	group membership: `x2=1` if patient belongs to Combination `x2=0` else
`covar`	difference of baseline UACR and the corresponding cut-off point
`modeltype`	either `u1` (BC log UACR as input) or `u2` (log UACR as input)

Details

The functions `compute.c` and `compute.c2` are both used to calculate the c-statistics for the logistic and longitudinal models. The difference between these two functions is that in `compute.c` the marker `x` is binary and in `compute.c2` the new marker consists of three groups (here randomization groups). If `x1` and `x2` are both set to zero, then the patient belongs to Telmisartan.
If `modeltype="u1"`, then the difference of baseline UACR and cut-off point on the Box-Cox log scale should be entered in `covar` and if `modeltype="u2"`, then the difference of baseline UACR and cut-off point on the log scale should be entered in `covar`.

Value

This function returns the c-statistics for the logistic and longitudinal model (`c.logist` and `c.lme`, respectively), as well as the correlation of predicted probabilities of progression and observed progression status of both models (`r.logist` and `r.lme`, respectively).

Examples

```
# example for compute.c:
# load functions: simdata, boxcox, invboxcox, progression, cutpoint and progr.lme
set.seed(239048)
library(mnormt); library(nlme)
simdat1<-simdata(n=100, gamma11=0.05)
fit1<-lme(logalb~time+x+time:x, random=~1+time|id, data=simdat1$longi)
fit2<-glm(progr~x+covar, family=binomial(link="logit"), data=simdat1$newdata)
simdat2<-simdata(n=100, gamma11=0.05)
compute.c(fitl=fit1, fitb=fit2, newdata=simdat2$newdata, modeltype="u2")

# example for compute.c2:
# load functions simdata, boxcox, invboxcox, progression, cutpoint and progr.lme2
set.seed(239048)
library(mnormt); library(nlme)
simdat1<-simdata(n=100, gamma11=0.05)
logbase<-invboxcox(x=simdat1$nbase)
x1<-sample(c(0,1),100,TRUE)
x2<-numeric(100)
for(i in 1:sum(x1==0)) if(x1[i]==0) x2[i]<-sample(c(0,1),1)
simdat1$longi$x1<-rep(x1,each=3)
simdat1$longi$x2<-rep(x2,each=3)
fit1<-lme(logalb~time+x1+time:x1+x2+time:x2, random=~1+time|id,data=simdat1$longi)
fit2<-glm(progr~x1 + x2+covar,family=binomial(link="logit"), data=simdat1$newdata)
compute.c2(fitl=fit1, fitb=fit2, n=100, x1=x1, x2=x2, progr= simdat1$newdata$progr,
 covar=simdat1$newdata$covar, modeltype="u2")
```

7.1.1.12 IDI function

Description

This function calculates the integrated discrimination improvements (IDI) for the logistic and the longitudinal models regarding a new marker.

Requirements

The function `compute.idi` requires two fit objects of the logistic models (one fit objects in which the new marker `x` is an independent variable in the logistic model and one without the marker `x`) and two fit objects of the longitudinal models (again with and without the marker `x` in the mod-

els), as well as a dataset. In the function compute.idi2 the new marker are the randomization groups, which are specified by the dummy variables x1 and x2.

It is also important to specify which model type was used to calculate the fit objects (either u1 for BCLOG models or u2 for LOG models).

Additionally, the self-written R functions invboxcox (see subsection 7.1.1.3)and progr.lme (see subsection 7.1.1.7) are required for these functions.

Usage

```
compute.idi(fitb, fitbx, fitl, fitlx, newdata, modeltype=NULL)
compute.idi2(fitb, fitbx, fitl, fitlx, covar, n, progr, x1, x2, logbase,
 modeltype=NULL)
```

Arguments

fitb	fit object of the logistic model without the new marker (i.e. x or x1 and x2)
fitbx	fit object of the logistic model with the new marker (i.e. x or x1 and x2)
fitl	fit object of the longitudinal model without the new marker (i.e. x or x1 and x2)
fitlx	fit object of the longitudinal model with the new marker (i.e. x or x1 and x2)
newdata	test sample
n	number of patients in the dataset
progr	specifies if the patient is (progr=1) or is not (progr=0) progressing in CKD
x1	group membership: x1=1 if patient belongs to Ramipril x1=0 else
x2	group membership: x2=1 if patient belongs to Combination x2=0 else
logbase	log urine albumin excretion value at baseline
modeltype	either u1 (BC log UACR as input) or u2 (log UACR as input)

Details

The function compute.idi2 calculates the IDI for the logistic and the longitudinal models for a new marker, which consists of three groups (here: randomization groups expressed by the dummies x1 and x2). If the new marker consists only of two groups, then use the function compute.idi.

If x1 and x2 are both set to zero, then the patient belongs to Telmisartan.

If modeltype="u1", then the difference of baseline UACR and cut-off point on the Box-Cox log scale should be entered in covar and if modeltype="u2", then the difference of baseline UACR and cut-off point on the log scale should be entered in covar. Regardless of the modeltype, UACR must be entered in logbase on the log scale.

Value

This function returns the IDI of the logistic and for the longitudinal model (`IDI_logstistic` and `IDI_longitudinal`, respectively).

Examples

```
# example for compute.idi:
set.seed(239048)
library(mnormt); library(nlme)
simdat1<-simdata(n=100,gamma11=0.05) # requires functionsIDI.r
fit1<-lme(logalb~time, random=~1+time|id,data=simdat1$longi)
fit1x<-lme(logalb~time+x+time:x, random=~1+time|id,data=simdat1$longi)
fit2<-glm(progr~covar, family=binomial(link="logit"), data=simdat1$newdata)
fit2x<-glm(progr~x+covar, family=binomial(link="logit"), data=simdat1$newdata)
simdat2<-simdata(n=100,gamma11=0.05)
compute.idi(fitl=fit1, fitlx=fit1x, fitb=fit2,fitbx=fit2x, newdata=simdat2$newdata,
modeltype="u2")

# example for compute.idi2
set.seed(239048)
library(mnormt); library(nlme)
simdat1<-simdata(n=100,gamma11=0.05) # requires functionsIDI.r
logbase<-invboxcox(x=simdat1$nbase)
x1<-sample(c(0,1),100,TRUE)
x2<-numeric(100)
for(i in 1:sum(x1==0)) if(x1[i]==0) x2[i]<-sample(c(0,1),1)
simdat1$longi$x1<-rep(x1,each=3)
simdat1$longi$x2<-rep(x2,each=3)
fit1<-lme(logalb~time, random=~1+time|id,data=simdat1$longi)
fit1x<-lme(logalb~time+x1+time:x1+x2+time:x2, random=~1+time|id,data=simdat1$longi)
fit2<-glm(progr~covar,family=binomial(link="logit"), data=simdat1$newdata)
fit2x<-glm(progr~x1 + x2 + covar,family=binomial(link="logit"),
data=simdat1$newdata)
compute.idi2(fitl=fit1,fitlx=fit1x,fitbx=fit2x, fitb=fit2, n=100, x1=x1, x2=x2,
 progr=simdat1$newdata$progr, covar=simdat1$newdata$covar, logbase=logbase,
 modeltype="u2")
```

7.1.2 R functions for the analysis of the ONTARGET-SysKid study

In this subsection we present a description for all self-written R functions which were used for the investigation of the ONTARGET-SysKid dataset. These functions are: `winsor`, `boxcox3`, `boxcox4`, `inBCwinsor`, `target`, `progr.lme2`, `compute.c2` and `compute.idi2`.

7.1.2.1 Winsor function

Description

The function `winsor` ensures that all log UACR are positive. All values under -2.281208 are set to -2.281208 and then all values are shifted by 2.281209. This threshold is the 1 % percentile of the ONTARGET-SysKid dataset ($n = 5293$).

Usage

```
winsor(x)
```

Arguments

`x`	log UACR; UACR in mg/mmol

Value

Winsorized log urine albumin excretion values

Example

```
set.seed(40580)
loguacr<-invboxcox(x=rnorm(100))
hist(winsor(loguacr))
```

7.1.2.2 Boxcox function

See self written R functions `boxcox3` and `boxcox4` in subsection 7.1.1.1.

7.1.2.3 Inverse function to winsor and boxcox

Description

The function `invBCwinsor` is the inverse function to the functions `boxcox3` and `winsor`.

Usage

```
invBCwinsor(lambda=0.67,x,mue=1.041583,sigma=1.17939)
```

Arguments

`x`	BC log UACR values
`lambda`	Box-Cox parameter λ
`mue`	mean of Box-Cox transformed values
`sigma`	standard deviation of Box-Cox transformed values

Details

Please note that a value for `x` below -2.14 produces NAs.

The parameters for the `invBCwinsor` functions were optimized manually on the ONTARGET-SysKid dataset in order to fulfill the assumption of normally distributed residuals in the longitudinal models. For details, please see 20121019_BCoptim_manually.r and optimalBCparameters_forONTARGET.r in the supplementary files of the appendix.

Value

Log urine albumin excretion values

Example

```
set.seed(93041)
norm<-rnorm(1000)
hist(invBCwinsor(x=norm))
```

7.1.2.4 Target function

Description

The `target` function - for the optimization of the Box-Cox parameter λ - was defined as the standard deviation of the difference between standardized residuals of a longitudinal model and the quantiles of a standard normal distribution.

Requirements

The self-written R function `boxcox4` is required for this function.

Usage

```
target(lambda, x, time, ID, ret=T, show.plot=F)
```

Arguments

`lambda`	Box-Cox parameter λ
`x`	log urine albumin excretion values
`time`	time of measurements in years
`ID`	identification of patients
`ret`	determines the output
`show.plot`	determines whether qqplots are returned or not

Details

By default `ret = TRUE`. This means that the standard deviation of the difference between standardized residuals of a longitudinal model and the quantiles of a standard normal distribution, as well as the mean and the standard deviation of the Box-Cox transformed data are returned. If `ret = FALSE`, only the standard deviation of the difference between the quantiles of the standard normal distribution and the Box-Cox transformed data is returned.

The parameter `show.plot` is by default FALSE to suppress qqplots.

Value

The mean (`mean_boxcox`) and the standard deviation (`sd_boxcox`) of the Box-Cox transformed data, as well as the standard deviation between the difference of the quantiles of a standard normal distribution and the quantiles of the Box-Cox transformed data (`sd_qq`) are returned.

Example

```
# load the functions boxcox variant 2, invboxcox, cutpoint, progression and simdata
library(mnormt)
library(nlme)
set.seed(349058)
testdata<-simdata(n=1041, gamma11=0.05)
# load boxcox4 and winsor function
loguacr<-winsor(testdata$longi$logalb)
target(lambda=0.5, x=loguacr, time=testdata$longi$time, ID=testdata$longi$id,
 show.plot=T, ret=TRUE)

# optimize the Box-Cox parameter lambda:
result<-optimize(f=target, interval=c(0,3), ret=F, x=loguacr,
 time=testdata$longi[,"time"], ID=testdata$longi[,"id"], maximum=F)
result
```

7.1.2.5 Probability of progression function

See self-written R function `progr.lme2` in subsection 7.1.1.7.

7.1.2.6 C-Statistic Function

See self-written R function `compute.c2` in subsection 7.1.1.11.

7.1.2.7 IDI function

See self-written R function `compute.idi2` in subsection 7.1.1.12.

7.2 Probability of Progression Plots

In this subsection the probability of progression plots are shown for each and every simulation scenario of this Master thesis (see subsection 4.2). The probability of progression was estimated from both models, i.e. the logistic and longitudinal, by assuming 100 equally spaced baseline urine albumin excretion values which ranged from 0.001 to 33.79 mg/mmol. All coefficients except for γ_{11} were assumed to be constantly zero ($\gamma_{00}= 0$, $\gamma_{01}= 0$ and $\gamma_{10}= 0$). The black line always represents the true probability of progression, the green line the probability of progression estimated by the longitudinal model and the red line the probability of progression estimated by the logistic model.

7.2.1 No optimization of BC parameters

Figures 34 to 57 show the estimated probability of progression of the simulation scenarios of subsection 4.2.2.1, i.e. using global Box-Cox parameters in the BCLOG models.

7.2.1.1 Figures for group x = 0

Figures 34 to 37 show the progression probabilities for group $x = 0$ in the BCLOG and LOG models. Recalling the longitudinal model definition, we see that these plots are identical for each γ_{11}. Therefore they are only shown for one γ_{11}. Figure 34 shows the mean and median probability of progression over 1000 simulation runs, which were estimated by logistic and longitudinal BCLOG model.

Figure 34.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 0; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

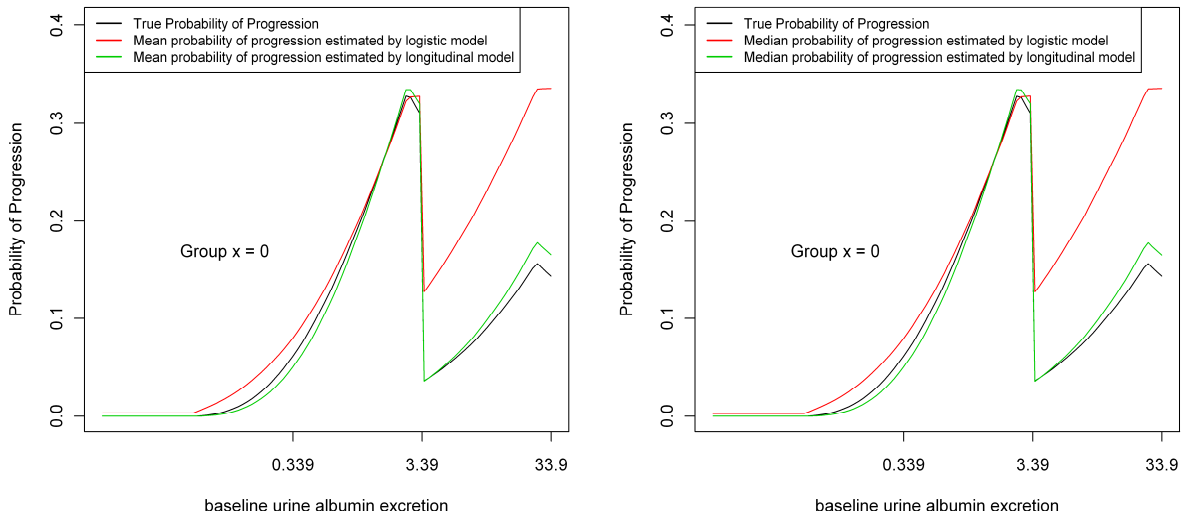


Figure 35 illustrates the first and third quartiles of probability of progression over 1000 simulation runs of both models of interest. The left panel shows the real quartiles, whereas the scaling of the vertical axis in the right panel is the arc sin square root transform of the probabilities in order to enlarge differences with very low probabilities.

Figure 35.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 0; left panel: 25% and 75% percentiles over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentiles over 1000 simulations

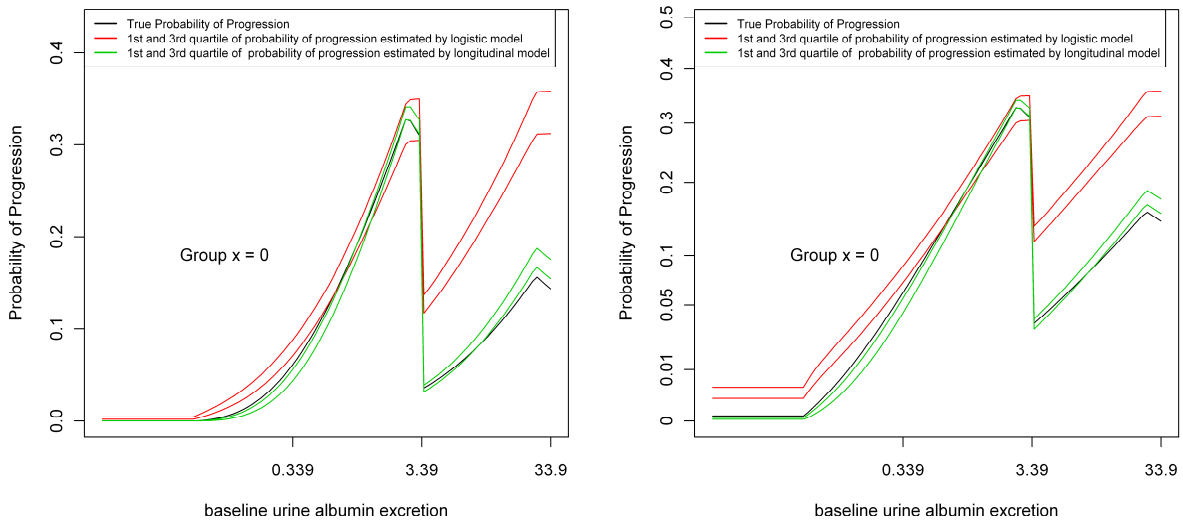


Figure 36 illustrates the mean (left panel) and median (right panel) probability of progression over 1000 simulation runs estimated by the logistic and longitudinal LOG models. Figure 37 shows the untransformed (left panel) and arc sin square root transformed (right panel) first and third quartiles of the probability of progression over 1000 simulation runs by the two LOG models of interest.

Figure 36.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 0; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

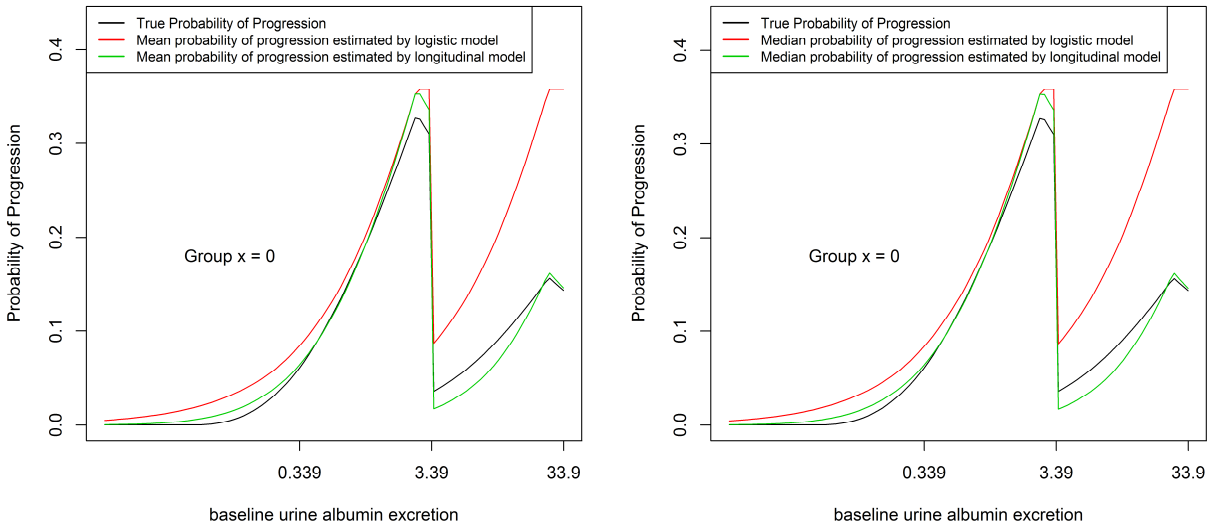
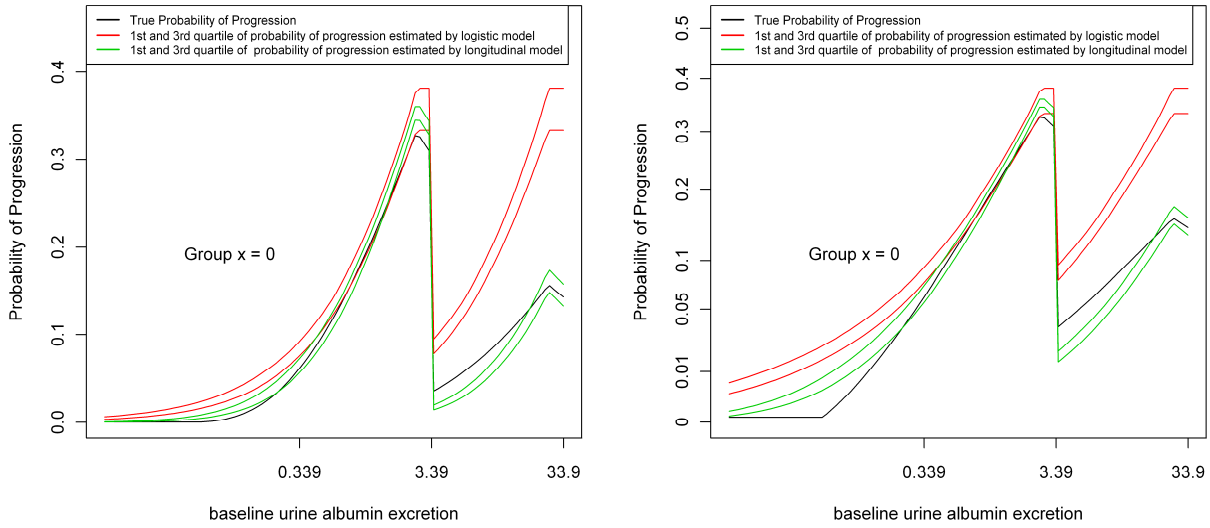


Figure 37.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 0; left panel: 25% and 75% percentiles over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentiles over 1000 simulations



All these Figures (34 to 37) show that the longitudinal models estimate the probability of progression for patients from group $x = 0$ with more precision than the logistic models. This is the case for both, the BCLOG and LOG models.

7.2.1.2 Figures for group x = 1

Now we consider only fictive patients of group $x = 1$. Figures 38 to 57 show the true and estimated probabilities of progression for this group. The slope of group $x = 1$ in the longitudinal model, γ_{11}, was varied from 0 to 0.05 by steps of 0.01. Since the probability of progression does not differ for group $x = 0$ or group $x = 1$ if γ_{11} is set to 0, the Figures are only shown once in this Master thesis (see Figures 34 to 37).

Figures 38, 42, 46, 50 and 54 illustrate the mean (left panel) and median (right panel) probability of progression over 1000 simulation runs estimated by both BCLOG models of interest, i.e. logistic and longitudinal, for the above mentioned values of γ_{11}.

Figures 39, 43, 47, 51 and 55 show the untransformed (left panel) and arc sin square root transformed (right panel) first and third quartiles of the probability of progression over 1000 simulation runs estimated by the logistic and longitudinal BCLOG for the above mentioned values of γ_{11}.

Figures 40, 44, 48, 52 and 56 illustrate the mean (left panel) and median (right panel) probability of progression over 1000 simulation runs estimated by both LOG models of interest, i.e. logistic and longitudinal, for the above mentioned values of γ_{11}.

Figures 41, 45, 49, 53 and 57 show the untransformed (left panel) and arc sin square root transformed (right panel) first and third quartiles of the probability of progression over 1000 simulation

runs estimated by both LOG models of interest, i.e. logistic and longitudinal, for the above mentioned values of γ_{11}.

Figure 38.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11} = 0.01; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

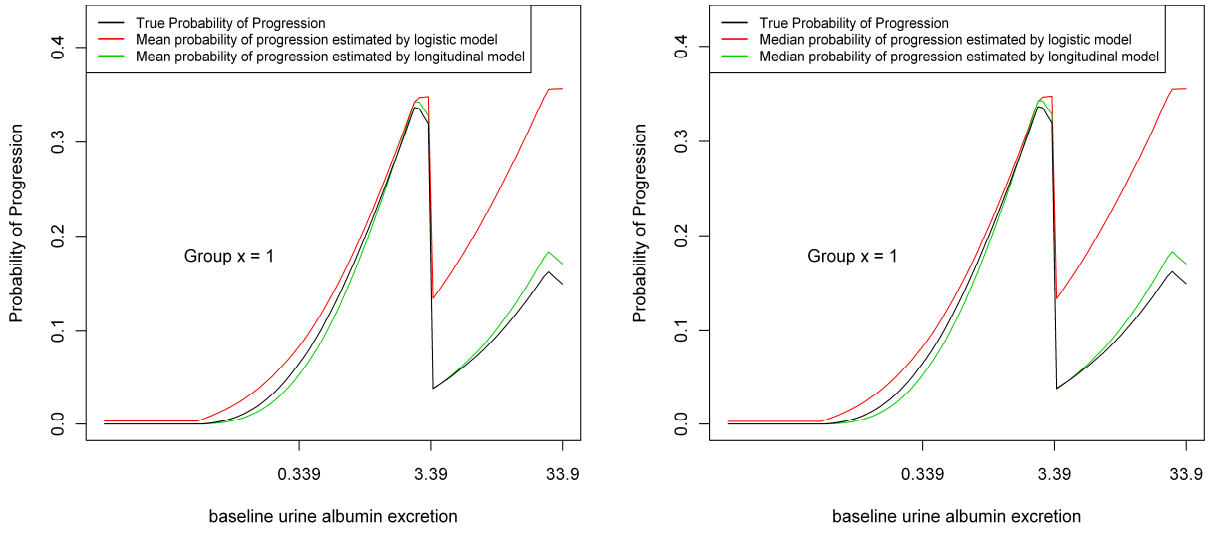


Figure 39.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 andγ_{11} = 0.01; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

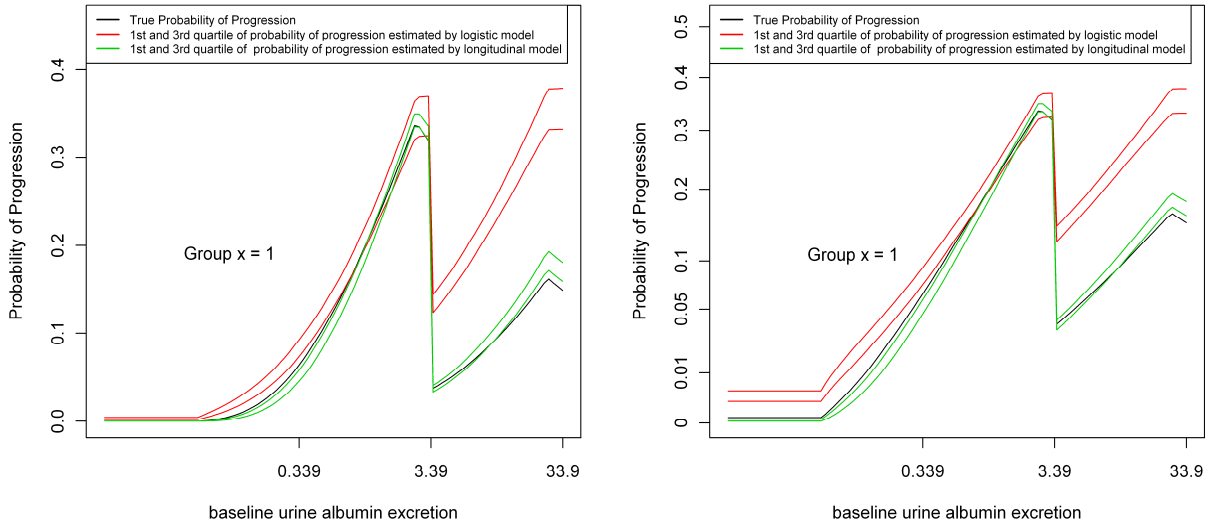


Figure 40.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 and γ_{11} = 0.01; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

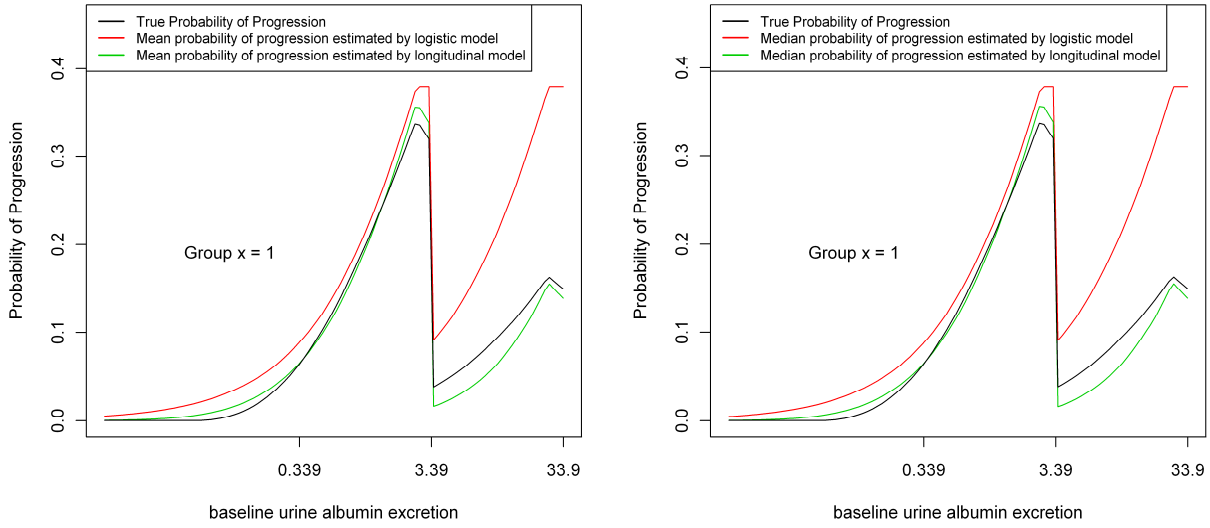


Figure 41.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 andγ_{11} = 0.01; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

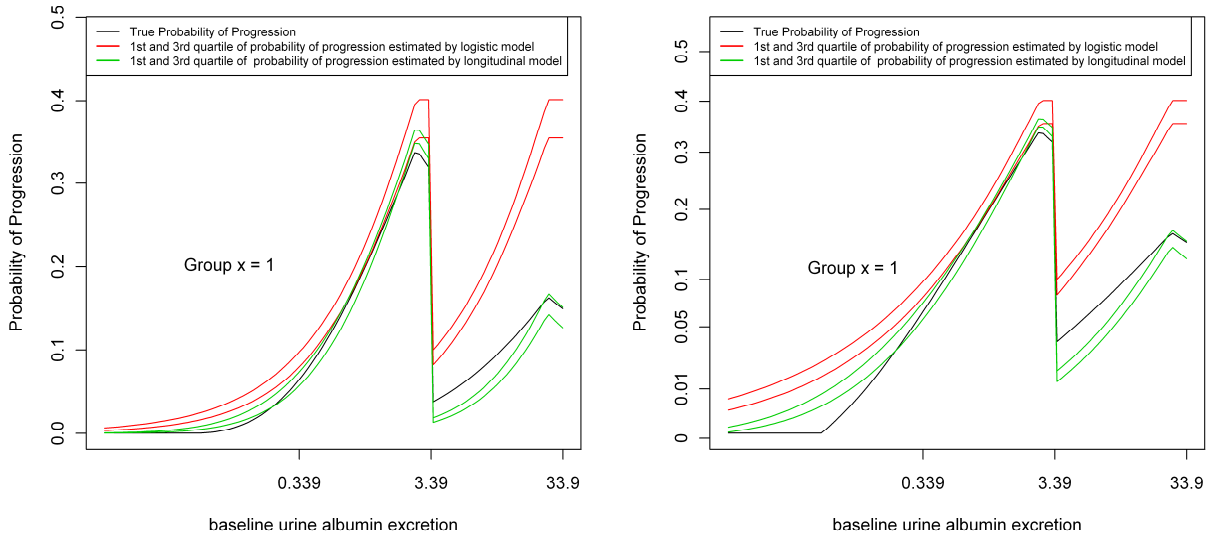


Figure 42.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11} = 0.02; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

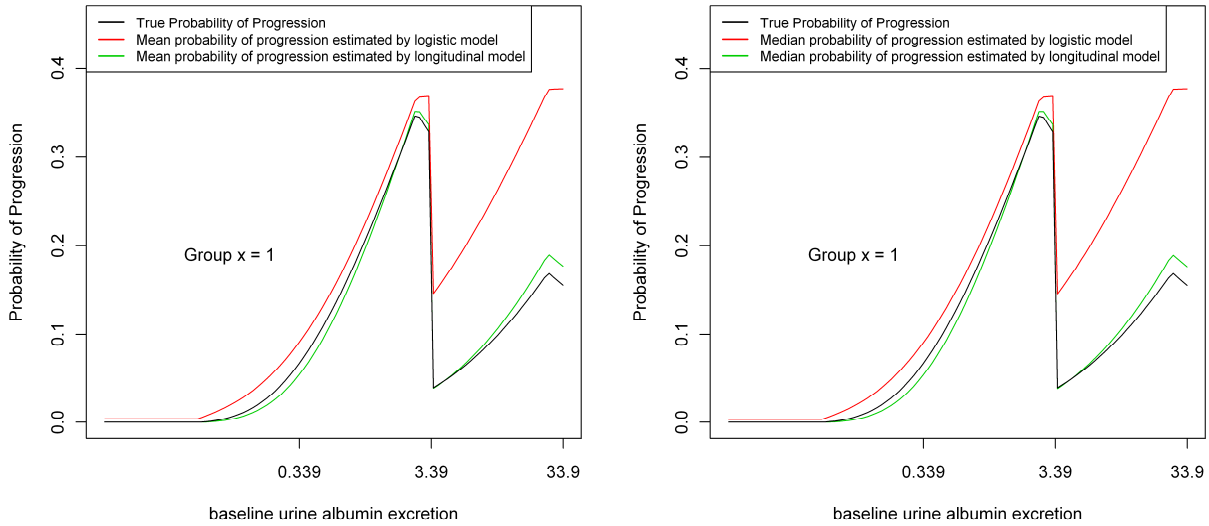


Figure 43.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 andγ_{11} = 0.02; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

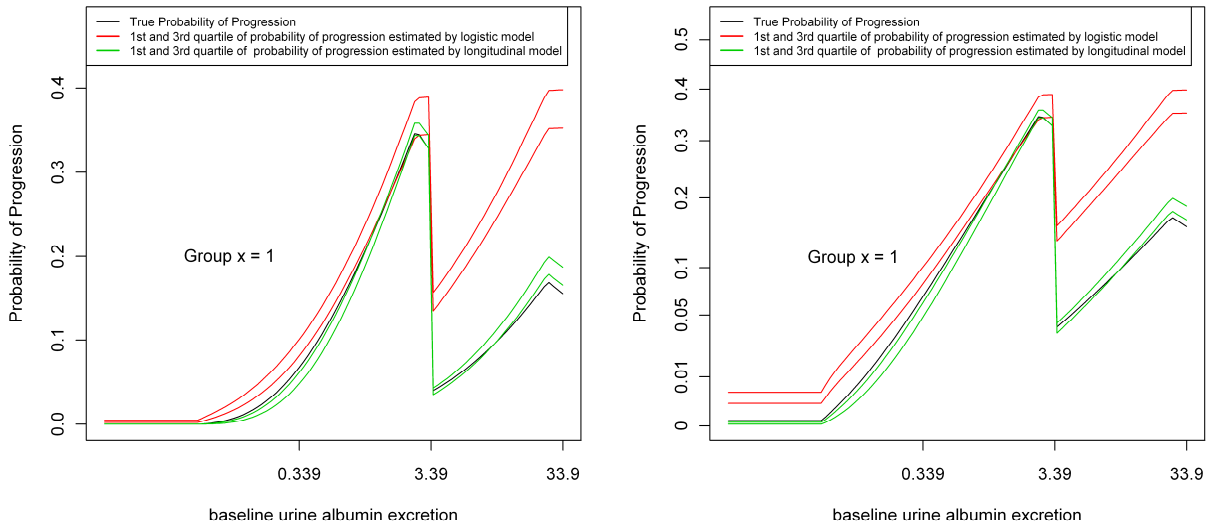


Figure 44. Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 and γ_{11} = 0.02; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

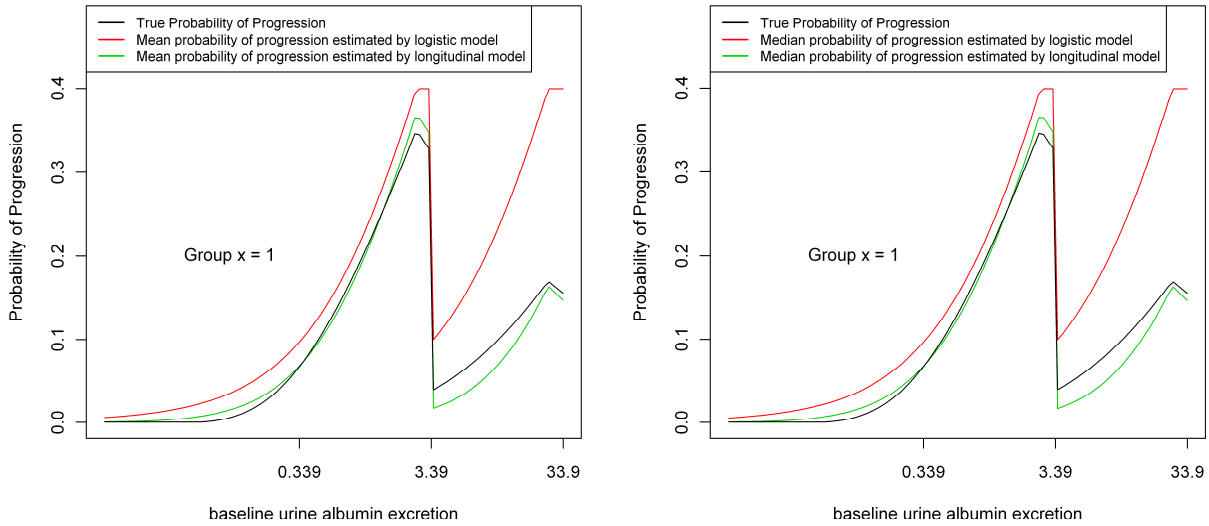


Figure 45. Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 andγ_{11} = 0.02; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

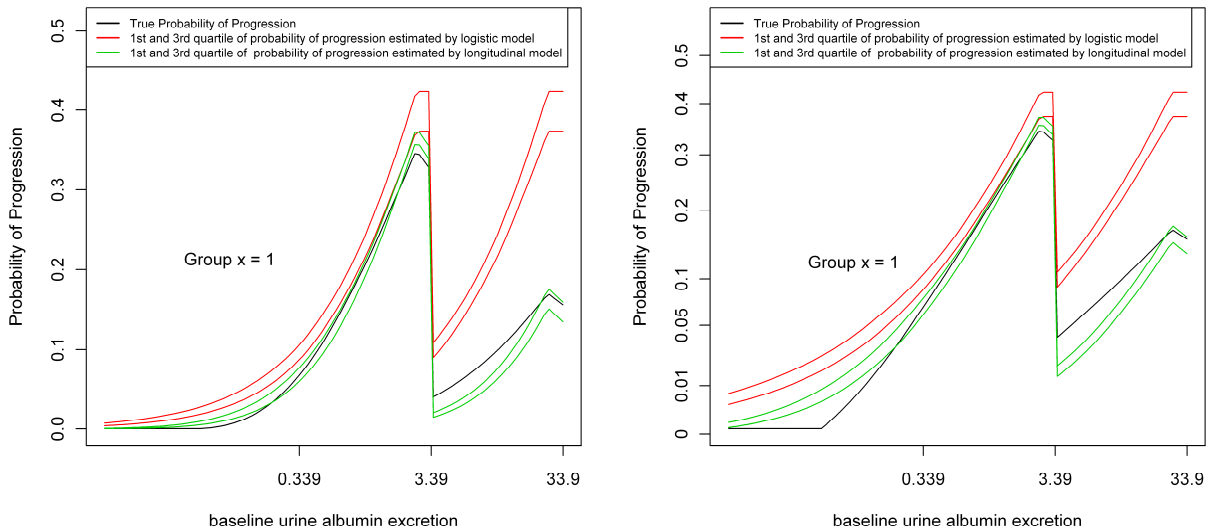


Figure 46.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11} = 0.03; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

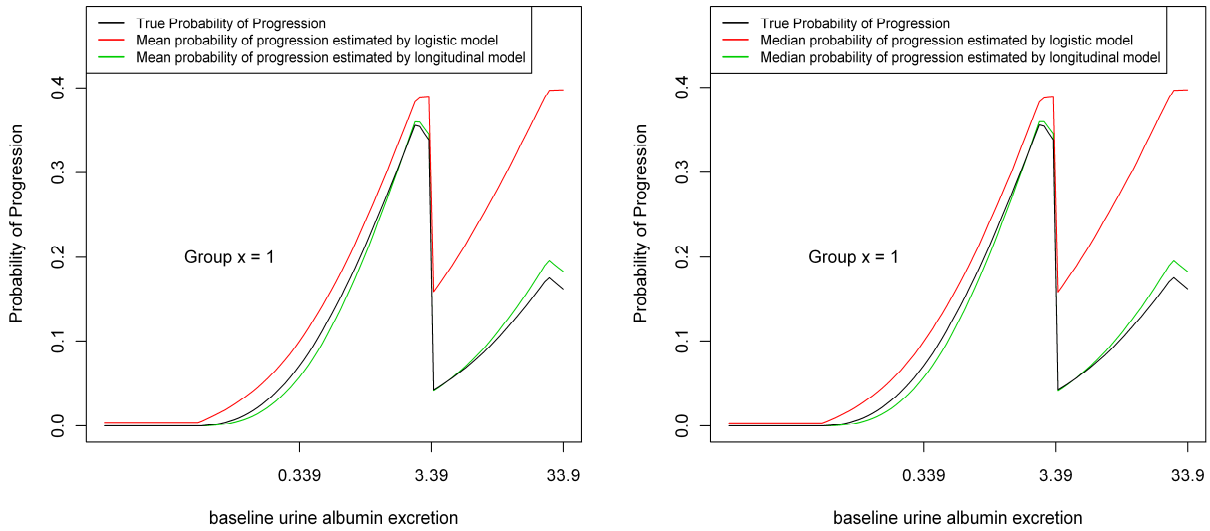


Figure 47.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 andγ_{11} = 0.03; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

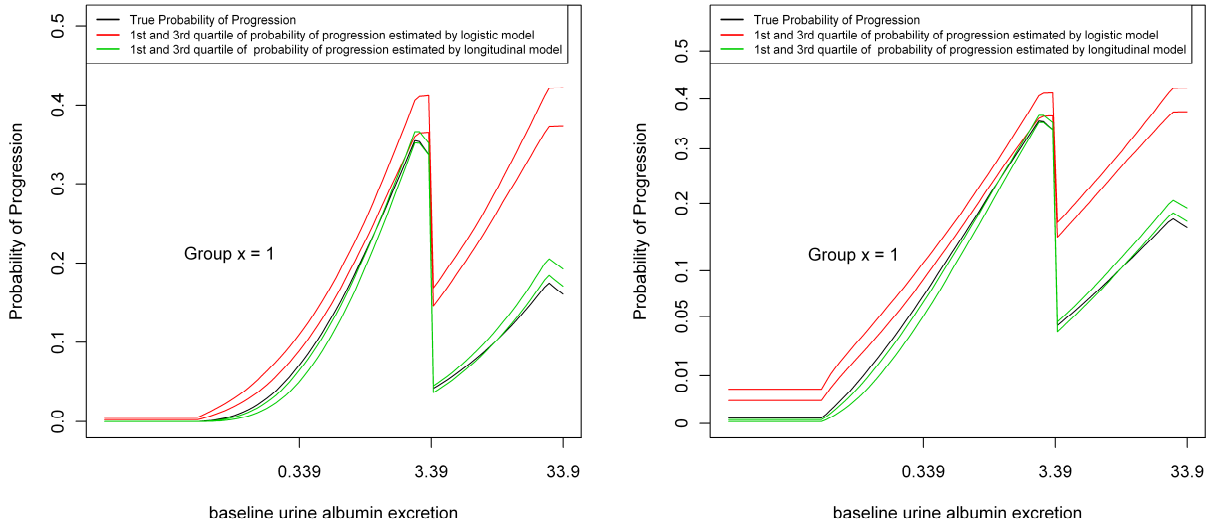


Figure 48.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 and γ_{11} = 0.03; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

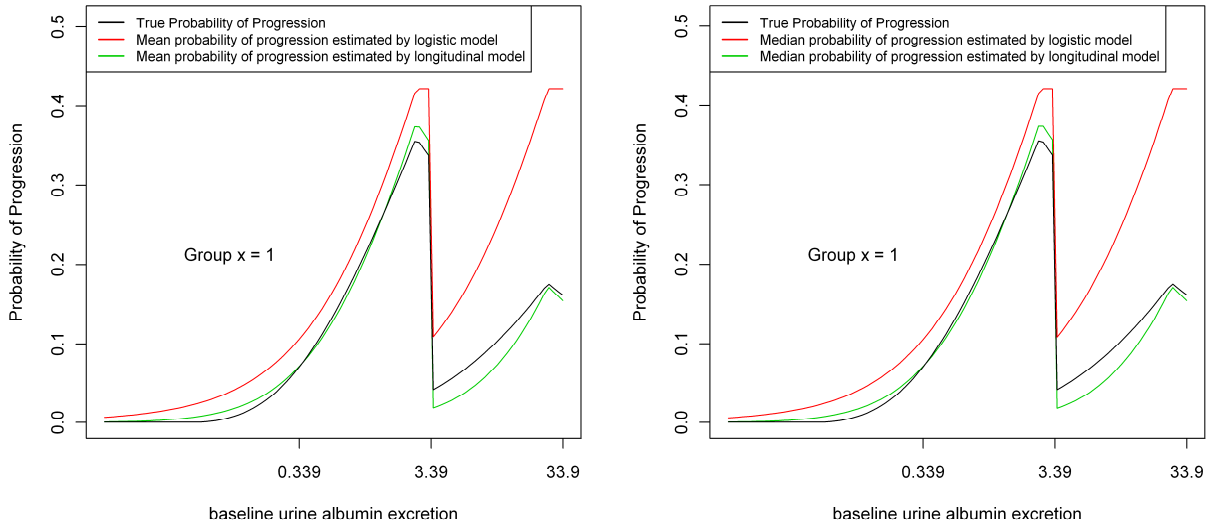


Figure 49.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 andγ_{11} = 0.03; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

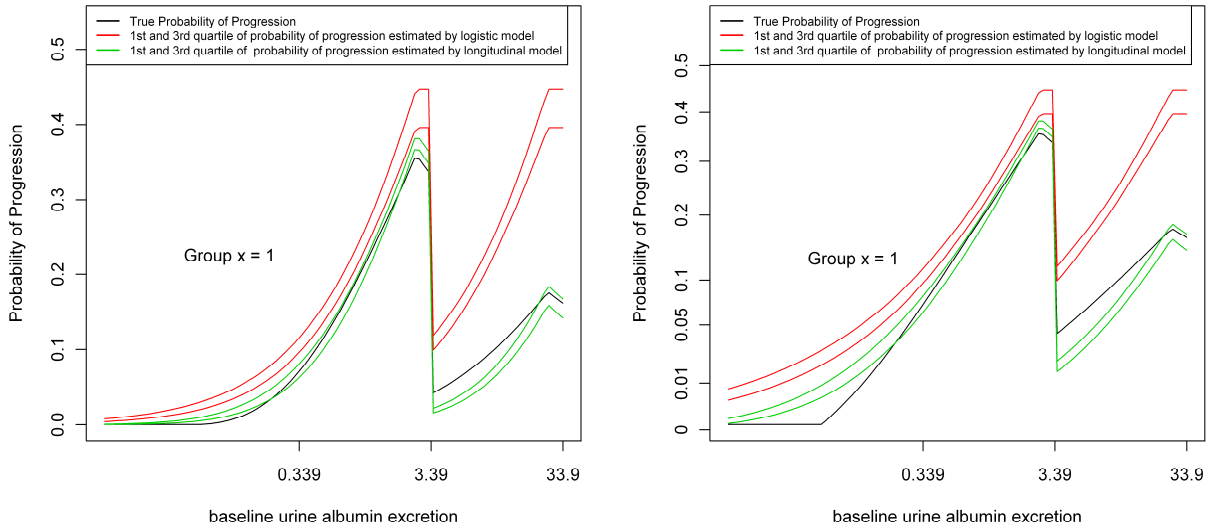


Figure 50.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11} = 0.04; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

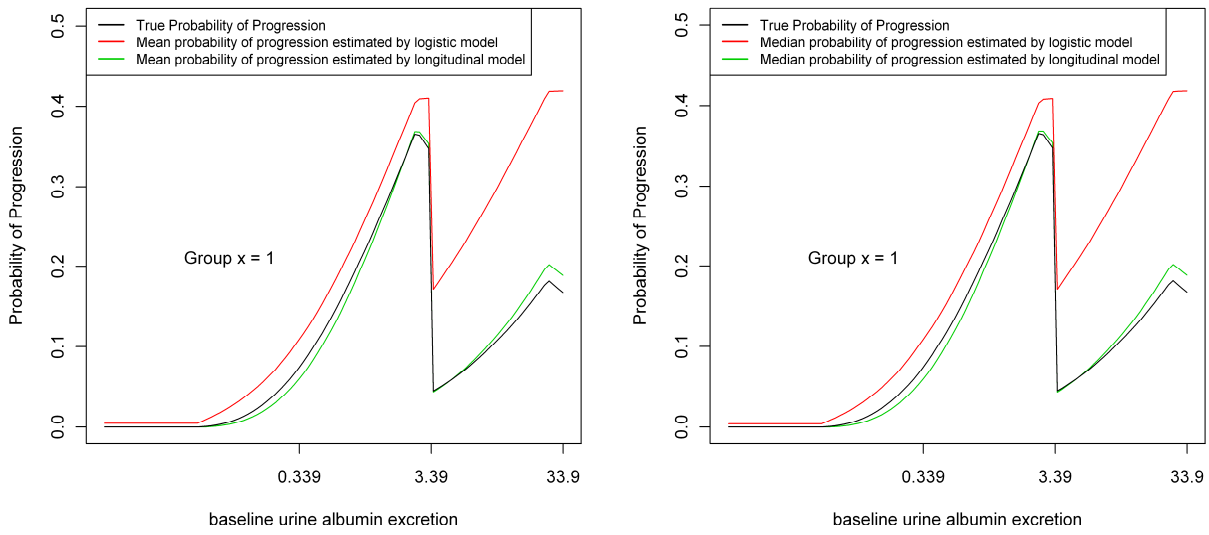


Figure 51.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 andγ_{11} = 0.04; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

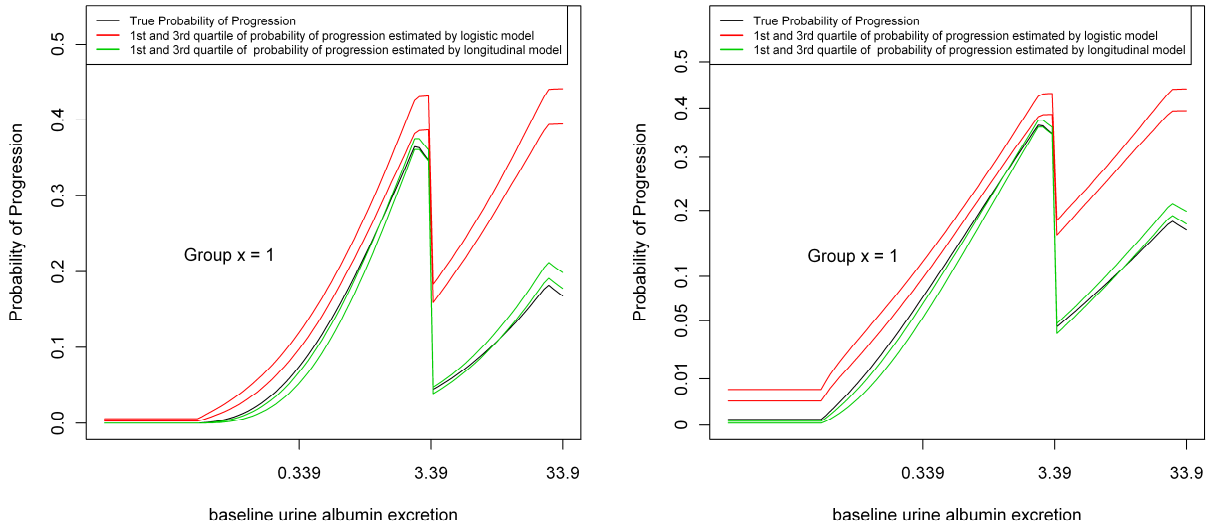


Figure 52.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 and γ_{11} = 0.04; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

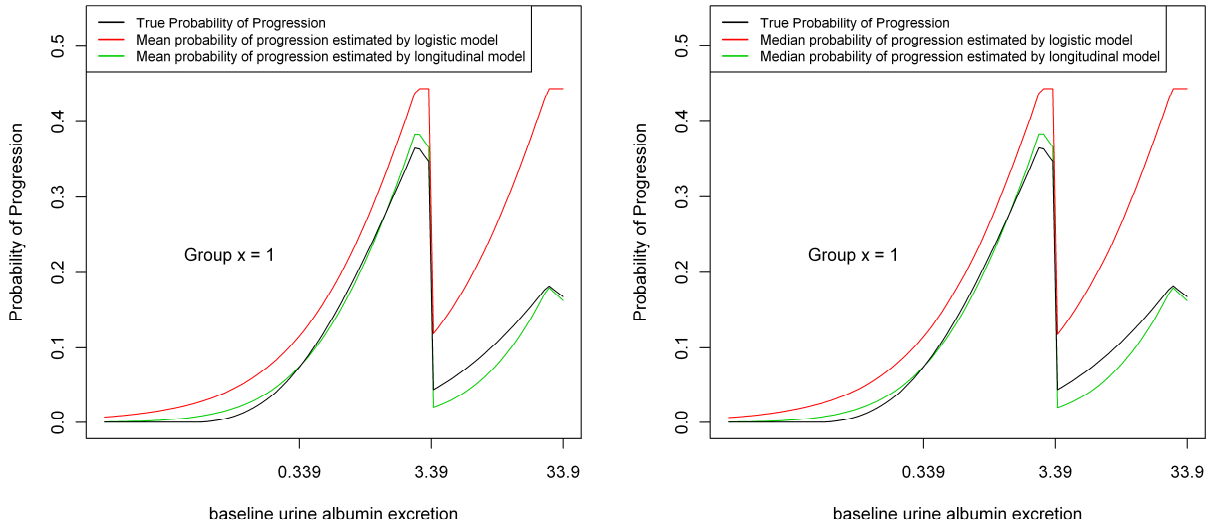


Figure 53.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 andγ_{11} = 0.04; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

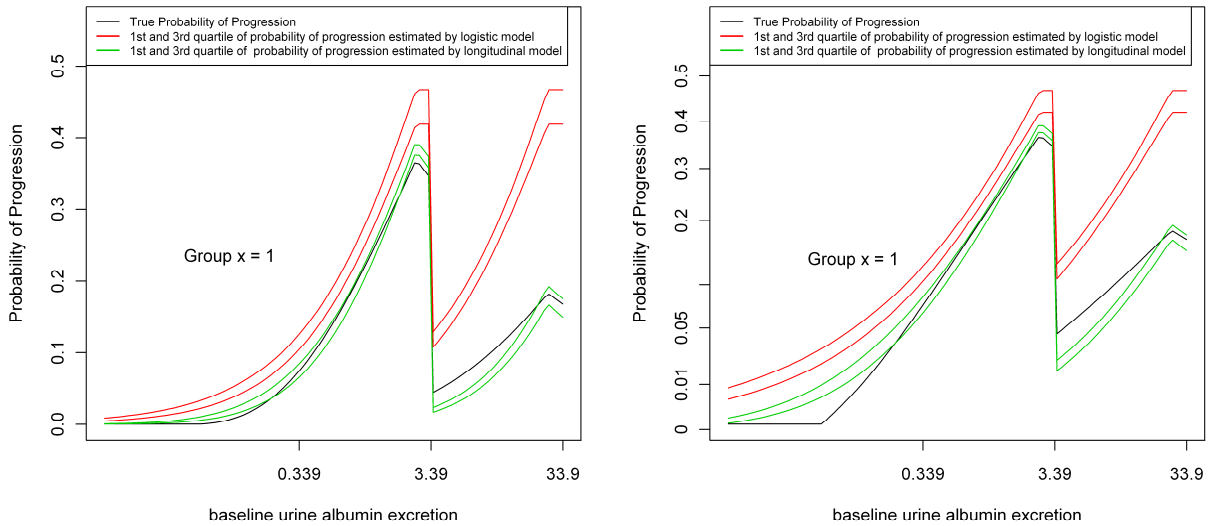


Figure 54.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11} = 0.05; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

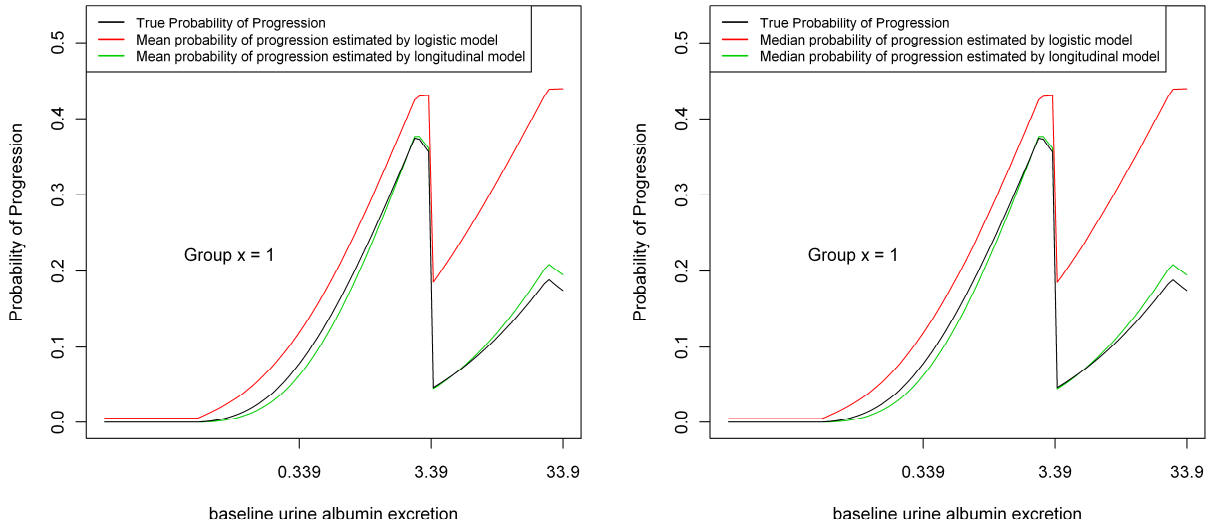


Figure 55.Probability of progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 andγ_{11} = 0.05; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations

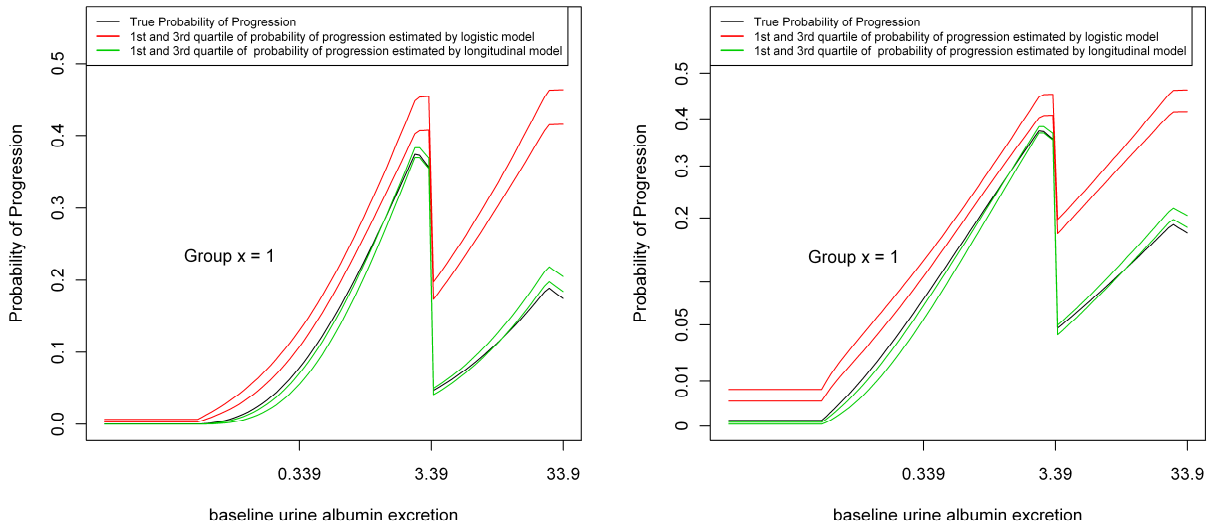


Figure 56.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 and γ_{11} = 0.05; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

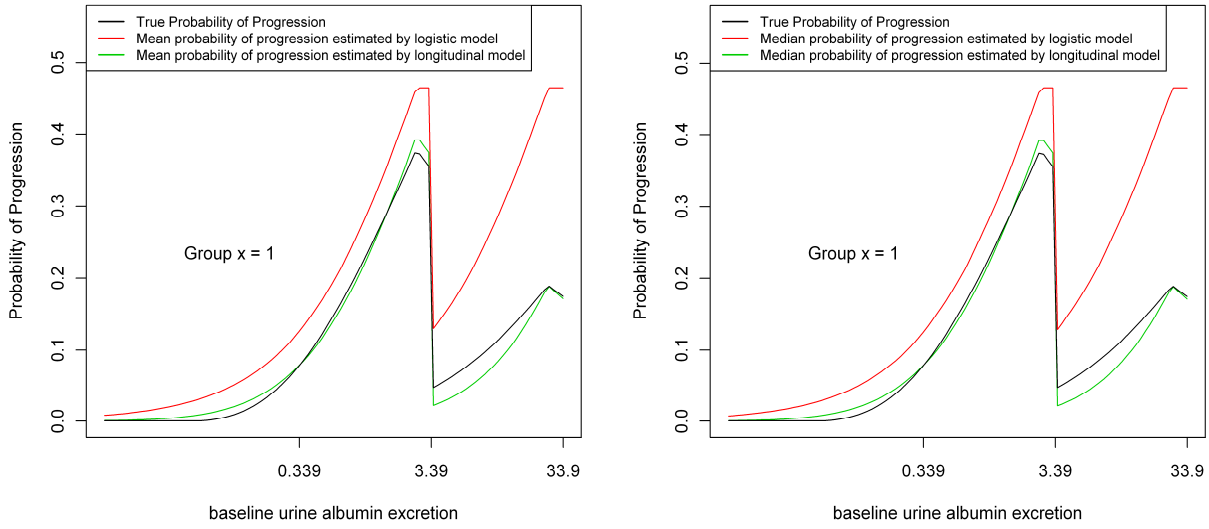
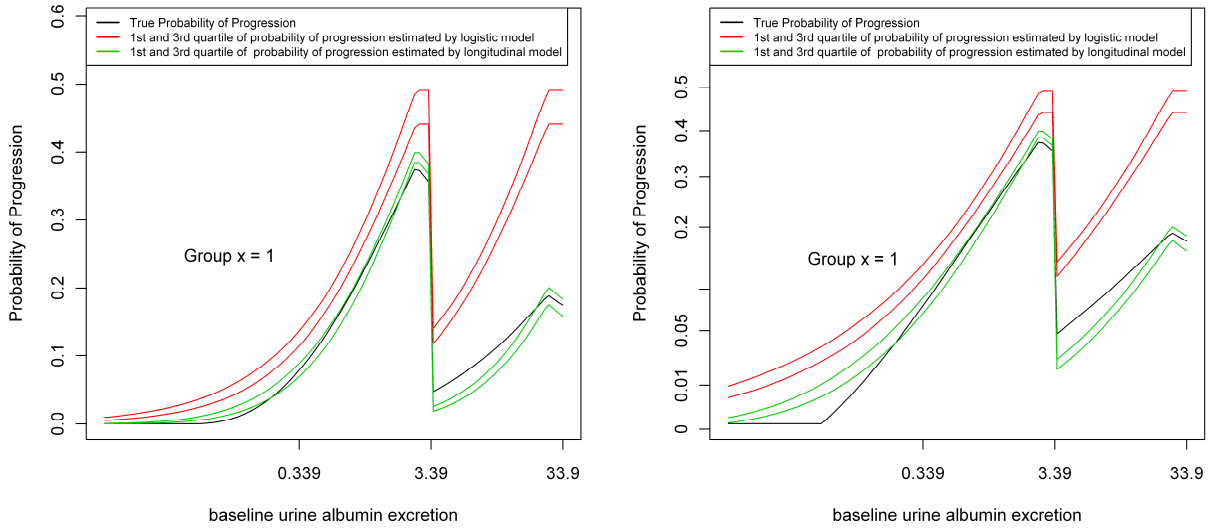


Figure 57.Probability of progression: true (black), estimated by the logistic LOG model (red), estimated by the longitudinal LOG model (green); x = 1 andγ_{11} = 0.05; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations



All these probability of progression plots (Figures 34 to 57) show that the logistic model always overestimates the probability of progression and that the estimation by the longitudinal model is much closer to the true probability of progression than the estimation by the logistic model. This is the case for both groups of x, all γ_{11} ranging from 0 to 0.05 and both model types, i.e. BCLOG and LOG. Furthermore, we notice a higher variability in the estimation of the probability of progression in the logistic models than in the longitudinal models.

If γ_{11} is increased all progression probabilities increase, whereat the probability estimated by the logistic model increases fastest and hence raises the estimation bias in these models with each γ_{11}.

In the prognosis of incidence, the logistic and longitudinal model are more or less equally good for group $x = 0$. Only for very small baseline urine albumin excretion values, the logistic model slightly over-estimates the probability of incidence, i.e., for normoalbuminuric patients at baseline. But when we consider prognosis of progression (for microalbuminuric patients) we notice that the logistic model considerably over-estimates the probability of progression. The first and third quartiles of the estimation of the probability of progression by the logistic model are far away from covering the true probability of progression. For group $x = 1$, we start with the same situation for $\gamma_{11} = 0$. As γ_{11} increases, we see that the estimation of the probability of progression worsens in the logistic model. The goodness of estimation by the longitudinal model is not affected by the increase of γ_{11}. The estimation of the probability of progression by the longitudinal model is always much closer to the true probability of progression than the estimation by the logistic model.

The main difference between the estimations by BCLOG model and by LOG model is that in the LOG models the logistic model over-estimates not only the probability of progression, but also the probability of incidence. Again, this effect worsens by increasing γ_{11}. Furthermore, in the LOG models the probability of progression is slightly underestimated by the longitudinal model.

7.2.1.3 Different sample sizes

Since the Figures of the probability of progression do not differ for different sample sizes, they are not repeated at this point. For $\gamma_{11} = 0.05$, $n = 520$ and group $x = 0$ see Figures 34 to 37 and for $\gamma_{11} = 0.05$, $n = 520$ and group $x = 1$ see Figures 54 to 57. For $\gamma_{11} = 0.02$, $n = 2082$ and group $x = 0$ see Figures 34 to 37 and for $\gamma_{11} = 0.02$, $n = 2082$ and group $x = 1$ see Figures 42 to 45.

7.2.2 Optimization of BC parameters

Figures 58 to 63 show the estimated probability of progression of the logistic and longitudinal models of the simulation using local Box-Cox parameters and setting γ_{11} to 0.02 and 0.05, respectively. In this subsection we concentrate on BCLOG models only; see subsection 7.2.1 for the LOG models.

7.2.2.1 Figures for group x = 0

Figures 58 and 59 show the results for patients of group $x = 0$. Recalling the longitudinal model definition, we see that these plots are identical for each γ_{11}. Therefore, they are only shown once. Figure 58 shows the mean (left panel) and median (right panel) probability of progression estimated by both BCLOG models of interest, i.e. logistic and longitudinal, over 1000 simulation runs for the above mentioned values of γ_{11}.

Figures 59 illustrates the untransformed (left panel) and arc sin square root transformed (right panel) first and third quartiles of the probability of progression over 1000 simulations estimated by the logistic and the longitudinal BCLOG model for the above mentioned values of γ_{11}.

Figure 58.Probability of Progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 0; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

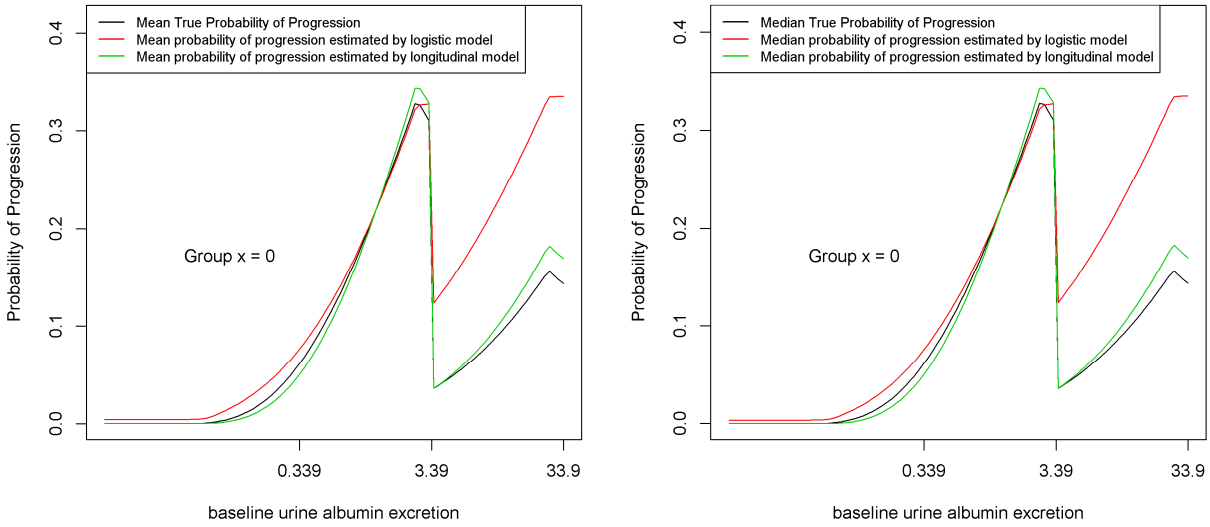
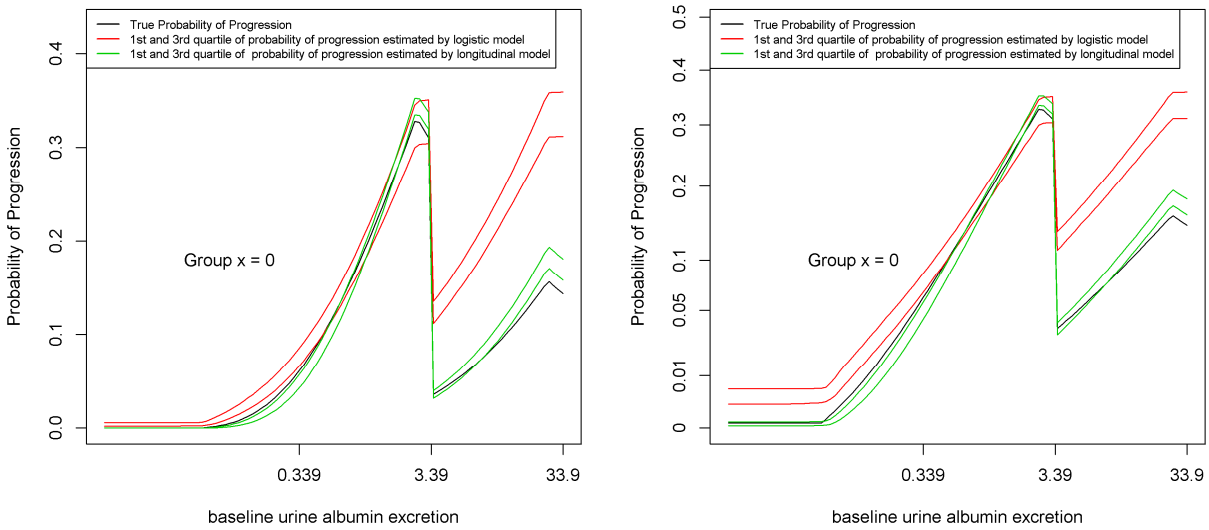


Figure 59.Probability of Progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 0; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations



No relevant difference can be detected when comparing the probability of progression estimations using local Box-Cox parameters (Figures 58 and 59) to those using global Box-Cox parameters in the models (Figures 34 and 35).

7.2.2.2 Figures for group x = 1

Figures 60 to 63 show the true and estimated probabilities of progression for fictive patients of group $x = 1$ and γ_{11} equal to 0.02 and 0.05, respectively. Figures 60 and 62 illustrate the mean (left panel) and median (right panel) and Figures 61 and 63 the untransformed (left panel) and arc sin square root transformed (right panel) first and third quartiles of the probability of progression estimated by both BCLOG models of interest, i.e. logistic and longitudinal, over 1000 simulations.

Figure 60.Probability of Progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11}= 0.02; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

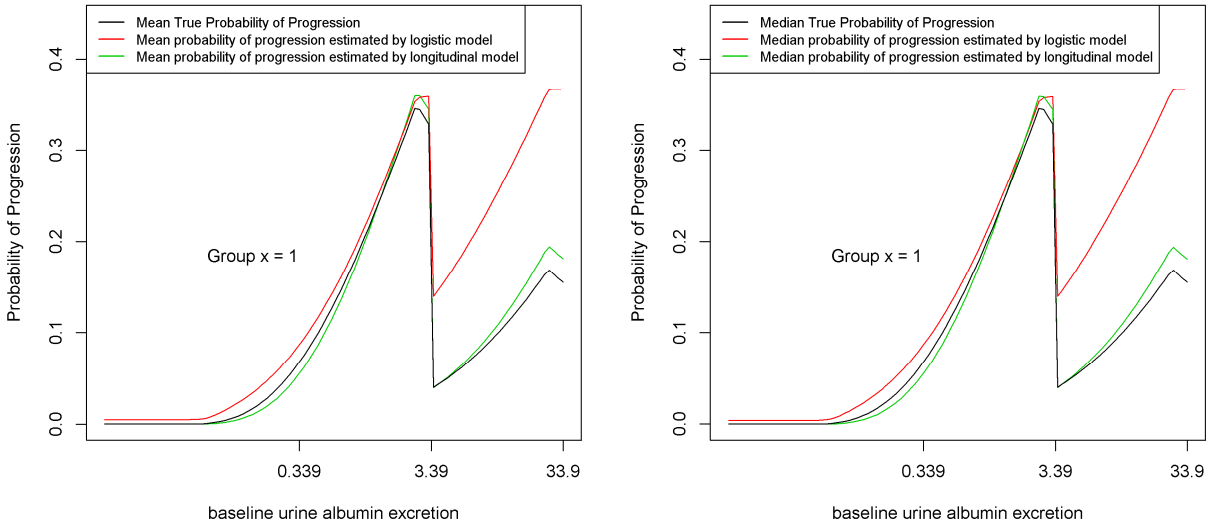
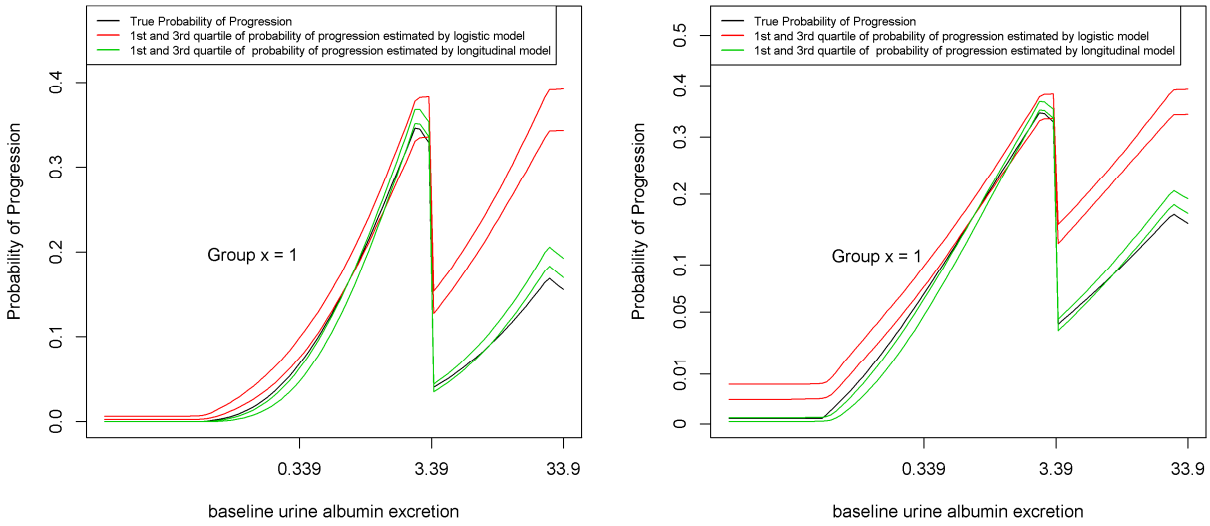


Figure 61.Probability of Progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11}= 0.02; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations



Comparing Figures 60 and 61 to Figures 42 and 43 reveals on the one hand, a slightly higher bias in estimation of the probability of progression by the longitudinal model (especially in the prediction of incidence) when applying the local Box-Cox parameters instead of the global ones in the models. On the other hand, the probability of progression estimation by the logistic model is to some extent better (especially in the prediction of incidence since the 1st and 3rd quartiles over 1000 simulation runs of this probability encloses the true probability of progression near the first cut-off point), when using local Box-Cox parameters instead of global ones for the transformation of log UACR. Nevertheless, the prediction by the longitudinal model is still better than by the logistic model.

Figure 62.Probability of Progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11}= 0.05; left panel: mean over 1000 simulations; right panel: median over 1000 simulations

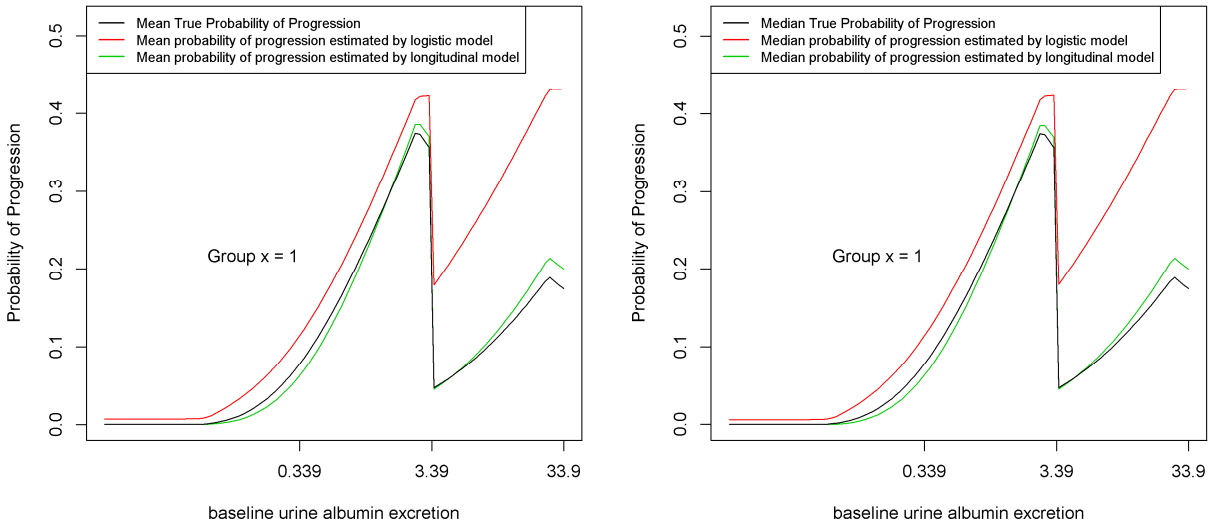
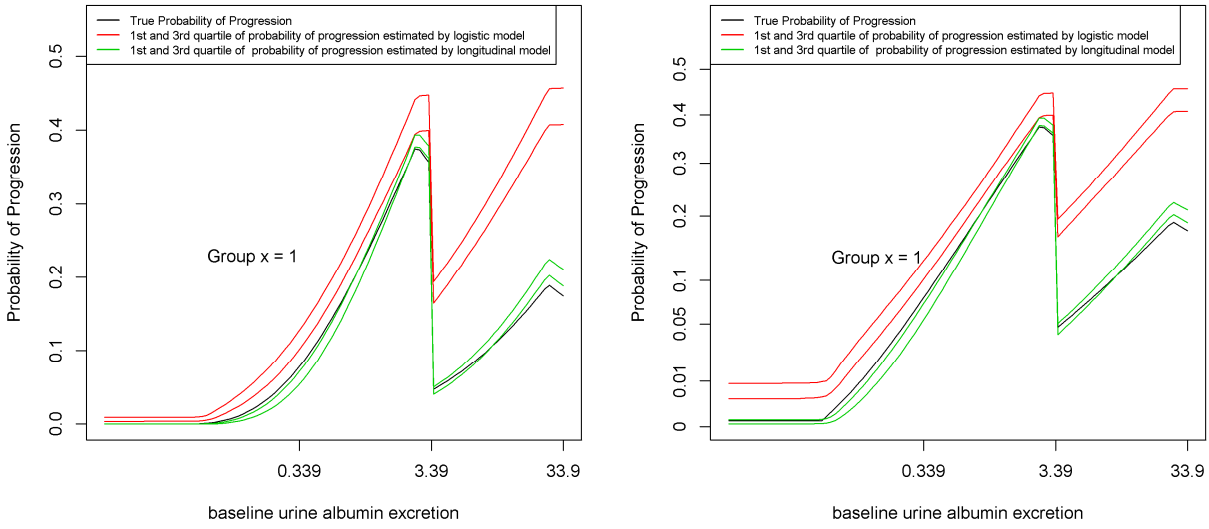


Figure 63.Probability of Progression: true (black), estimated by the logistic BCLOG model (red), estimated by the longitudinal BCLOG model (green); x = 1 and γ_{11}= 0.05; left panel: 25% and 75% percentile over 1000 simulations; right panel: arc sin square root transformed 25% and 75% percentile over 1000 simulations



When comparing Figures 62 and 63 to Figures 54 and 55 we observe no difference in the estimation of the probability of progression by the logistic model. However, the probability of progression estimation by the longitudinal model is slightly worse (especially for the prediction of incidence) when applying the local Box-Cox parameters instead of the global ones in the transformation of the log UACR. Nonetheless, the overall prediction of progression by the longitudinal model is still better than the one by the logistic model. Like before, the variability in the estimation of the probability of progression is higher in the logistic models than in the longitudinal models (Figures 60 to 63).

7.3 Supplementary files

UACR transformations

20140508_UACR_scales.r
20110810_Transformation.txt
ontarget600.txt
Figure5.r
BoxCox+Results+(13.01.2012).r.txt
BoxCox+Results+(29.08.2012).r

Other preparations

20120119_findingGmatrix.r
20120119_Gmatrix+residual error.r
Results+(20-01-2012).r.txt
20120208_Trajectoryoffictivep.r
20120323_FunctionsforSim_withoutoptim_with Georg.r
20140510_ProbProgrTrueFunction.r
20120726_Tables.xlsx
20111102_Simwithcovar.txt

Sample size for simulation

20120427_determinesaplesize_forbatch.r
20120427_detn_diff_seeds.bat
20120516_nopt_plots.r

self-written R functions

all_functions_for_package.r
functions.r
functions_opt.r
functionsIDI.r

No optimization of BC parameters

sim_part1.r
sim_part2.r
sim_part3.r
20121130_Sim_withoutoptim_results.xlsx
20120726_Tables.xlsx
20120723_Results.r
20120725_allPlots_probofprogr.r
20140512_density_trueProb.r
20140719_coefficients_longitudinal_model.r
sim_beta_0_1000.Rdata to sim_beta_0.05_1000.Rdata
simtablec_0_1000.Rdata to sim_tablec_0.05_1000.Rdata
simplotsc_0_1000.Rdata to simplotsc_0.05_1000.Rdata
IDIaC.r
IDI_CStat_0.02_0.03.bat
IDI_CStat_0.05_0.01.bat
IDI_CStat_0_0.04.bat
IDIaC_u1_0_1000.Rdata to IDIaC_u2_0.05_1000.Rdata
20120725_testsample_results.r
20120723_ProbabilityPlots.r
sim_beta_0.05_520_1000.Rdata
sim_beta_0.02_2082_1000.Rdata
simtablec_0.05_520_1000.Rdata
simtablec_0.02_2082_1000.Rdata
simplotsc_0.05_520_1000.Rdata
simplotsc_0.02_2082_1000.Rdata
IDIaC_u1_0.02_1000_2082.Rdata
IDIaC_u1_0.05_1000_520.Rdata
IDIaC_u2_0.02_1000_2082.Rdata
IDIaC_u2_0.05_1000_520.Rdata

Optimization of BC parameters

simopt_part1.r
simopt_part2.r
simopt_part3.r
20120801_FunctionsforSim.r
20120817_results_optim.r
20130221_Sim_optim_results.xlsx
20121005_plots_for_probability_of_progr.r
simopt_0.02_1000.Rdata and simopt_0.05_1000.Rdata
simtable_0.02_1000c.Rdata and simtable_0.05_1000c.Rdata
simplots_0.02_1000c.Rdata and simplots_0.05_1000c.Rdata
20121010_IDIopt_results.r
IDIaCopt_u1_0.02_1000.Rdata
IDIaCopt_u1_0.05_1000.Rdata
IDIaCopt_u2_0.02_1000c.Rdata
IDIaCopt_u2_0.05_1000.Rdata
t.test.r

Analysis of the ONTARGET-SysKid study

functions_for_Daniela2.r
20121019_BCoptim_manually.r
optimalBCparameters_forONTARGET.r
Maria_Oct_1.RData
winsor.r
20120927_Comparison_on_real_data.r
Maria_Oct_2.RData
Maria_Oct_3.RData
121026_boxplot.modelu1.pdf
121026_boxplot.modelu2.pdf
20130719_descriptive_analysis.r
Maria_July.RData
histtotalUACR.png
histtotallogUACR.png
histlogUACRgroups.baseline.png
histlogUACRgroups.2years.png
histlogUACRgroups.5years.png
histtotalBClogUACR.png
histBClogUACRgroups.baseline.png
histBClogUACRgroups.2years.png
histBClogUACRgroups.5years.png
20121109_descri_results.r
20140720_resultsplots.r

7.4 Curriculum Vitae

Maria Neubauer, BAKK. BA

Education

Since 10/2011	**University of Vienna,** statistics Magister programme
10/2008–08/2014	**University of Vienna,** final degree in sociology Bachelor programme Bachelor paper: *"Gibt es Unterschiede in der Freizeitgestaltung und dem Bildungsinteresse zwischen PensionistInnen mit und ohne körperlichen Einschränkungen bei Alltagstätigkeiten?"* Field of application: Questionnaire evaluation Research internship: "Gesundheit und Arbeit im Lebenslauf" Field of application: Questionnaire evaluation
10/2008-07/2011	**University of Vienna,** final degree in statistics Bakk. programme Bakkalaureat papers: • *„Maximierung des Ernteertrags von Bauernhof XY“* Field of application: Linear optimization • *„Hängt die Persistenz der Volatilität von den Frequenzen ab?“* Field of application: Time series analysis
09/2003–06/2008	**HAK I** in Wels International stream with special focusing on languages and marketing A level in English, Mathematics, Marketing, Russian, German and commercial studies A level's project: *„We proudly present Tae Kwon Do“* Responsible for: Public relations, design of advertisement materials, event organization, documentation und presentation
09/1995–07/2003	**Elementary and secondary modern school** in Neumarkt/H.

Scholarships

2008–2014	Endowment Fund of the Julius-Raab-Stiftung
2011–2013	Performance Scholarship of the University of Vienna

Conferences

10/2014	Predictive Analytics Conference
08/2014	Annual Conference of the International Society for Clinical Biostatistics
05/2014	IDC Data Hub Conference

Work experience

Since 11/2012	Volunteering **Caritas Vienna, Project Le⁺O** Responsible for: Issuing the identification card, giving out food
07/2011–06/2013	Member of Student Council, 2. deputy **Students Council for Statistics at the University of Vienna** Responsible for: Consultation of enrolment, representative at BeSt, mandatory in WIWI study conference, mandatory in the postdoctoral lecture qualification committee, organisation of study travels
07–09/2011	Project team member **Medical University of Vienna, Institute of Biometry** Responsible for: Statistical analysis and reporting of two projects
05–06/2011	Call Centre Agent **TRICONSULT Wirtschaftsanalytische Forschung GmbH** Responsible for: Implementation of telephone surveys

Membership

Since 01/2013	Österreichische Statistische Gesellschaft, Forum Junge Statistik

Computer skills

MS-Office	User skills
Perl	Basic knowledge
Statistical programmes	• **R:** 5 years programming experience • **SAS:** Course "Programmierung 1" at SAS Austria • **SPSS:** User skills

Language skills

German	Mother tongue
English	Proficient user language study travel to Malta (2004) and Canterbury, GB (2005)
French	Basic user language study travel to Cannes, FR (2006)
Spanish	Basic user language study travel to Málaga, ES (2007)
Russian	Basic user language study travel to Moscow, RU (2011)
Slovak	Basic user summer college in Nitra, SK (2013)
Italian	Basic user summer college in Bovec, SI (2013)