Computationally Fast Approaches to Genome-Wide Association Studies

Karolina Sikorska

2014

Printed by: Ipskamp Drukkers
Cover design: Ruud Terhaag

Computationally Fast Approaches to Genome-Wide Association Studies

Snelle algoritmen voor
genoombrede associatiestudies

Proefschrift

ter verkrijging van de graad van doctor aan de
Erasmus Universiteit Rotterdam
op gezag van de rector magnificus

Prof.dr. H.A.P. Pols

en volgens besluit van het College voor Promoties

De openbare verdediging zal plaatsvinden op
woensdag 24 september 2014 om 9:30 uur

door

Karolina Sikorska

geboren te Bydgoszcz, Polen

Promotiecommissie

Promotoren:	Prof.dr. E. Lesaffre Prof.dr. A.G. Uitterliden Prof.dr. P.J.F. Groenen
Co-Promotor:	Dr. F. Rivadeneira
Overige leden:	Prof.dr. C. van Duijn Prof.dr. G. Verbeke Prof.dr. F.A. van Eeuwijk

Contents

Chapter 1

Introduction

In this chapter we provide introductory information on genome-wide association studies. We briefly discuss the statistical methods used in the association analyses. Both genetical and statistical parts are described in a non-technical way to help the reader understand basic concepts related to our research. Furthermore, we introduce the motivating data set and explain the aims of this thesis.

1.1 Genome-wide association studies

There are many ways to investigate the relationship between genes and human traits. Exploring genetic patterns in families by linkage analysis, was successful in explaining monogenic diseases, controlled by a single gene. However, common medical conditions, such as diabetes or obesity, are associated with multiple loci each having a small effect. The motivation to explain complex diseases together with the large progress in genotyping technology, made it possible to focus on genome-wide association studies (GWAS). The major ingredient in GWAS is the single nucleotide polymorphism (SNP), a genome position at which two (but rarely more) distinct nucleotide residues occur in at least 1% of the population. For instance, two sequenced DNA fragments from different individuals: AATGCTA and AATTCTA, contain one SNP: T replacing G at the middle fourth position. Such alternative forms of nucleotides are called alleles. Since individuals inherit a SNP allele from each parent, if we call the alleles A and B, the possible genotypes can be summarized as: AA, AB, BB. Since 2005, technological developments facilitated genotyping of over half a million SNPs in one go and at a reasonable price. In GWAS all SNPs are tested, one by one, for their association with the phenotype, which is either the susceptibility for the disease or the level of a trait. Identified SNPs may pinpoint new biological pathways, helping researchers to understand the mechanisms among the molecules in a cell. The expected effect sizes are rather small, usually below 0.5% contribution to R^2 in linear regression models and odds ratios below 1.2 in logistic regression models. On the other hand, to avoid false positive results, conservative multiple testing correction (Bonferroni) is applied to the commonly accepted significance level, leading to alpha level 5×10^{-8}. Those two factors together require very large sample sizes (thousands of individuals) to obtain reasonably small standard errors of the estimated coefficients.

The first large stream of the results published on GWAS dates back to 2007, when numerous papers were written on loci associated with (among others): diabetes, breast cancer, and lipids. From then onwards the field expanded enormously, investigating hundreds of human diseases and features, even choral singing (Morley et al., 2012), entrepreneurship (Van der Loos et al., 2010), and coffee drinking (Amin et al., 2012). All those endeavors are reported in the publicly available GWAS Catalogue (Hindorff et al., 2010), where over 1900 articles have been published by June 2014.

Effect sizes, quantified as variances explained or odds ratios, discovered in GWAS turned out to be even smaller than expected. A way to increase their significance is to enlarge the sample size even more. This led to meta-analyses performed across cohorts from all over the world. Different cohorts genotyping different SNPs hampered carrying out the pooled analyses. To be able to perform a meta-analysis became one of the driving forces behind the genotype imputation, using the fact that nearby SNPs are correlated. The dimensionality of

the data is increased by imputing genotypes that are not directly assayed, using the values of the two measured "tag" SNPs. Based on the HapMap project (Gibbs et al., 2003), the numbers of analyzed SNPs grew to approximately 2.5 million. Recently, with the advent of the 1000 Genomes Project (1000 Genomes Project Consortium and others, 2012) this number has even grown to 30 million.

The statistical tools used in association testing in GWAS belong to the set of popular and common methods. However, it is not the modeling that makes these studies challenging for statisticians. The necessity of fitting millions of models leads to computational struggles. In case of 30 million SNP data imputed using the 1000 Genomes Project, a GWA scan with simple linear regression takes a couple of days on a single computer. This time increases to weeks for logistic regression and to months or years for the linear mixed model. On the other hand, this "embarrassingly parallel" task can be split up, with little effort, among as many processors as desired/available. So called cluster-computing is indeed common in GWAS. Despite large-scale computing facilities, computing times remain long.

Another issue in GWA studies is caused by very large data files storing the SNP data. The total size of genotype data for 30 million SNPs reaches many hundreds of Gigabytes, stored in a few hundred files of a few Gigabytes each. Loading such large files, saved as ASCII files, demands a lot of time and computer memory. Moreover, the structure in which the data are stored also matters tremendously. Commonly, the data are written as "row per individual" in which all SNPs for an individual form a record on a disk. This structure is very inconvenient for a quick access of the selected SNPs of all individuals. In this thesis, we thoroughly discuss the problem of SNP data access. We show that it is a vital element of achieving fast speed of GWA computations. We overcome that problem using existing implementations.

We focus on cross-sectional continuous and binary outcomes as well as continuous longitudinal measurements. Below we describe statistical methods used in the analysis of this type of data.

1.2 Linear regression and generalized linear models

The major aim in statistics is to describe relationships between variables, which with some degree of realism reflect processes related to the subject of study. A formalization of those relationships is expressed in the form of statistical models. One of the oldest, the linear regression model, is used to formulate the association between a continuous outcome with a set of continuous or categorical predictors. Formally, the linear regression model for a set $\{y_i, x_{i1}, ..., x_{ip}\}_{i=1}^{n}$ has the following form:

$$y_i = \mathbf{x}_i'\beta + \epsilon_i \quad i = 1, ..., n, \tag{1.1}$$

where y_i denotes the response variable for individual i, $\mathbf{x_i}$ is a p-dimensional vector with predictors, $\beta = (\beta_1, ..., \beta_p)'$ is a p-dimensional vector with regression coefficients and ϵ_i represents measurement error that is assumed to be normally distributed. After stacking the n models (1.1) we obtain the vector form of the linear regression model

$$y = X\beta + \epsilon,$$

where y is $n \times 1$ dimensional vector, X is a $n \times p$ dimensional design matrix and ϵ is a n-dimensional vector. It is assumed that ϵ_i are normally distributed with mean 0 and standard deviation σ.

The coefficients in linear regression model are estimated using the ordinary least-squares method, which minimizes

$$||y - X\beta|| = \sum_{i=1}^{n} (y_i - \mathbf{x}_i'\beta)^2$$

with respect to β. The explicit solution is given by

$$\hat{\beta} = (X'X)^{-1}X'y.$$

The variance-covariance matrix of $\hat{\beta}$ is equal to

$$\sigma^2(X'X)^{-1}, \tag{1.2}$$

where σ^2 can be approximated by its unbiased estimator

$$s^2 = \frac{Y'Y - \hat{\beta}'X'X\hat{\beta}}{n-p}.$$

The standard error of $\hat{\beta}_j$ coefficient is the square root of the j-th diagonal element of the matrix in (1.2). Significance of the individual coefficients in the model is assessed using a t-test, because the statistic $\hat{\beta}_j/\text{se}(\hat{\beta}_j)$ has a t-distribution with $n-p$ degrees of freedom.

A flexible generalization of the linear regression model to other types of outcomes is given by the class of generalized linear models (GLM). There are three components building up the GLMs: the random component coming from the exponential family, the systematic component specifying the way the explanatory variables influence the response, and the link function defining the relationship between E(Y) and the systematic component. As an example consider the binary response with the model estimating the probability of "success". Every realization of response variable is treated as an outcome of the Bernoulli trial with $E(Y_i) = P(Y_i = 1) = p_i$. To prevent predicted probabilities to fall outside $[0; 1]$ the logit link function is used to relate p with the set of predictors. Namely, the model has a form

$$\log\left(\frac{p_i}{1-p_i}\right) = \mathbf{x}_i'\beta,$$

from which we can easily calculate the probability of an event as

$$p(\mathbf{x}_i) = \frac{\exp(\mathbf{x}_i\beta)}{1+\exp(\mathbf{x}_i\beta)}.$$

The model coefficients are typically estimated using the maximum likelihood (ML) method, which maximizes the logarithm of the joint probability mass function of a sample size n, i.e.,

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} p(\mathbf{x}_i)^{y_i}[1-p(\mathbf{x}_i)]^{1-y_i}.$$

Maximization is done using the Newton-Raphson or Fisher scoring method. These iterative procedures start with a tentative solution for the parameters and next improve it until convergence. In the Newton-Raphson method, the variance-covariance matrix of the parameters is obtained by inverting the negative of Hessian: a matrix with second order partial derivatives of the log-likelihood function. In the Fisher scoring approach the expected value of this matrix is inverted. Significance of the particular coefficients is assessed using the Wald test that evaluates the ratio $\hat{\beta}_j/\text{se}(\hat{\beta}_j)$, which has asymptotically a standard normal distribution.

1.3 Linear mixed models

One of the basic assumptions in linear regression is independence of the collected realizations of the random variable. In case of hierarchical data, this assumption is violated. An example of hierarchical, also called multilevel, structure are exam scores obtained from pupils within classes, classes within schools, and schools within districts. Measurements on individuals sharing a cluster are more similar than measurements collected in different clusters.

In this thesis we focus on a particular case of hierarchical data, namely longitudinal data. Here, the repeated measures collected on individuals over time form clusters of correlated outcomes. When the data are dependent, special care in the analysis is needed. Mixed models offer a flexible way of modeling correlated data. As given in Laird and Ware (1982) a mixed model describing the n_i-dimensional vector y_i of measurements collected on individual i over time is given by

$$y_i = X_i\beta + Z_i b_i + \epsilon_i, \quad i = 1, ..., n,$$

where X_i and Z_i are $n_i \times p$ and $n_i \times q$ design matrices with fixed and random effects. It is assumed that b_i has a multivariate normal distribution with zero means and variance-covariance matrix D, ϵ_i has a multivariate normal distribution with mean zero and variance-covariance matrix Σ_i. In case of independent errors with constant variance $\Sigma_i = \sigma^2 I_{n_i}$. Finally, it is also assumed that $b_1, ..., b_n, \epsilon_1, ..., \epsilon_n$ are mutually independent. The random effects describe the individual deviation from the average population evolution. They control for unobserved heterogeneity inducing the correlation between outcome variables from the same individual. The simplest structure of the random part consists of only the random intercept implying different baseline levels between individuals. This structure leads to a constant correlation over time, so called the compound symmetry structure. In many studies, this assumption is not realistic and therefore a random slope is added to allow for a variation in the individual slopes and thus implying a more complex correlation structure.

The estimation of variance components and fixed effects can be performed using the maximum likelihood method, which maximizes the log-likelihood

$$\log(L_{\text{ML}}) = -0.5 \sum_{i=1}^{n} \log |V_i| - 0.5 \sum_{i=1}^{n} (y_i - X_i\beta)' V_i^{-1} (y_i - X_i\beta) + c,$$

where $V_i = Z_i' D Z_i + \Sigma_i$ is the variance-covariance matrix of vector y_i. Maximization is done iteratively with respect to variance components in V_i and fixed parameters β. It has been shown that for small samples ML estimation can lead to biased estimates of $\hat{\sigma}^2$ (Verbeke and Molenberghs, 2009). Therefore, Restricted Maximum Likelihood (REML) method was proposed which maximizes the slightly modified log-likelihood

$$\log(L_{\text{ML}}) - 0.5 \log |\sum_{i=1}^{n} X_i' V_i^{-1} X_i|$$

providing an unbiased solution for $\hat{\sigma}^2$. The explicit solution for the fixed effects is given by

$$\hat{\beta} = \Big(\sum_{i=1}^{n} X_i' V_i^{-1} X_i \Big)^{-1} \sum_{i=1}^{n} X_i' V_i^{-1} y_i,$$

where the unknown variance components in V_i are replaced by their (RE)ML estimates and the variance-covariance matrix is given by

$$\text{var}(\hat{\beta}) = \Big(\sum_{i=1}^{n} X_i' V_i^{-1} X_i \Big)^{-1}.$$

A lot of attention has been spent on the proper estimation of the degrees of freedom in the inference for the fixed effects using an F or t-test. Several methods have been proposed by Satterthwaite (1941) and Kenward and Roger (1997), which may lead to slightly different p-values. This however concerns only situations with very small sample sizes or when linear mixed models are used outside the context of longitudinal data analysis (Verbeke and Molenberghs, 2009).

One way to estimate random effects is to use Bayesian ideas. The estimates called Empirical Best Linear Unbiased Predictors (EBLUPs) are given by

$$\hat{b}_i = D Z_i' V_i^{-1} (y_i - X_i \beta).$$

As provided by Henderson et al. (1959), for given variance components, the estimates for fixed and random effects can be obtained simultaneously by solving the following system of equations

$$\begin{pmatrix} X'\Sigma^{-1}X & X'\Sigma^{-1}Z \\ Z'\Sigma^{-1}X & Z'\Sigma^{-1}Z + \mathcal{D}^{-1} \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X'\Sigma^{-1}y \\ Z'\Sigma^{-1}y \end{pmatrix},$$

where X, y, b are obtained by stacking X_i, y_i, b_i, respectively, underneath each other. Furthermore, $\mathcal{D}, \Sigma$ and Z are block diagonal matrices with D, Σ_i and Z_i on the main diagonal and zeros elsewhere.

Another important issue related to linear mixed models are missing data. A popular classification as given by Little and Rubin (1989) distinguishes between: missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR). This classification is based on the relationship between the probability of an observation being missing and the previously collected or unobserved values of the outcome and covariates. The principles behind the missing data mechanisms are discussed in **Chapter 2**.

1.4 The Rotterdam Study and the genetics of bone loss

The Rotterdam Study is a population-based cohort prospective study which started in 1990 in the city of Rotterdam (Hofman et al., 2013). Initially 7983 individuals (so called RS-I cohort) were enrolled in the study. All participants were 55 years of age or over at baseline. This cohort was examined during 5 cycles: 1990-1993, 1993-1995, 1997-1999, 2002-2004 and 2009-2011 with sample sizes: 7983, 6315, 4797, 3550 and 2140 respectively. The objective of the Rotterdam Study is to explore factors that determine the occurrence of diseases frequent in the elderly populations such as: coronary heart disease, heart failure and stroke, Parkinson disease, Alzheimer disease, osteoporosis and other diseases. In over 1000 scientific articles and reports the researchers published on environmental, clinical and genetic risk factors of these diseases using Rotterdam Study data.

The part of the Rotterdam Study that inspired the research in this thesis relates to bone-strength in elderly people which is clinically assessed through bone mineral density (BMD) measurements. Low BMD is one of the strongest risk factors for osteoporotic fractures. Taking into account that osteoporotic fractures constitute an important public health problem, it is relevant to target their prevention. Identification of the genetic contribution is one of the routes to explaining BMD variation. A lot of work has been spent in investigating cross-sectionally measured BMD. Major GWAS findings report 56 loci influencing bone mineral density, which explains around 5% of its variation. Clinical speculations suggesting that some of those loci may be also related to age-related bone loss as well as conducted heritability studies motivated researchers to pursue genetic contributions to the BMD change (Mitchell and Yerges-Armstrong, 2011). Longitudinal studies are essential in relating the individual evolution of a trait with genetic characteristics. Measurements of BMD in the Rotterdam Study were collected 5 times for the RS-I cohort, however only 4 measurements were available by the time we conducted our research. The individuals were examined at baseline and on average after 2, 6, and 12 years. Unequal times of examinations and individuals missing their visit(s) generated an unbalanced data set.

1.5 Aim and outline of the thesis

Two major issues arise in performing genome-wide scans on longitudinal data. The first one is related to very small longitudinal SNP effects expected in GWAS. The sample sizes required to reach genome-wide significance level are very large and by now no longitudinal studies of this size have been conducted. The second challenge is linked to computational issues when millions of mixed models are to be fitted. Analyses of "longitudinal GWAS" demand sizeable computing resources and even then the computing time remains long.

This thesis focuses on the computational aspects of GWAS. We wish to show that large computing resources are not necessarily required in implementing genome-wide analyses. In addition, one should keep in mind that a statistical analysis is usually conducted multiple times by a researcher before the final result is obtained. The reasons for that include various factors such as: excluding/including subsets of individuals, correction for different sets of covariates (stratified or pooled analysis) or changes in the model. Also, purely technical factors, such as human errors during programming or power failure in the office facility,

might necessitate to repeat the analysis.

We propose methods which speed up the computations in GWAS by several orders of magnitude and facilitate whole genome scans on a single computer within reasonable time frames. The speedups are achieved by using approximations or by improving algorithms.

Particularly, in **Chapter 2**, we propose the conditional two-step (CTS) approach which simplifies the linear mixed model computations to estimating the regression coefficients in a simple linear regression. This procedure provides an approximation to the p-values for the longitudinal SNP effect. It involves the conditional linear mixed model, introduced by Verbeke et al. (2001). Our proposal is compared to other approximate procedures based on data reduction, which are known to us from the literature or collaborations with other researchers.

The CTS is further explored in **Chapter 3**. In this chapter we also consider the two-step approach which is a two-stage approach like the CTS, but now based on the classical linear mixed model. We focus in more detail on the theoretical fundaments of the approximate procedures for balanced situations. We describe results of an extensive simulation study comparing the performance of the two approximate methods in various data scenarios.

In **Chapter 4**, we focus on linear and logistic regression. For linear regression we propose an exact method, called semi-parallel regression, which estimates SNP effects, their standard errors and the p-values. This approach is based on replacing loops with large matrix operations, efficiently solving the least-squares problem for many SNPs at the same time. Many rearrangements in the computational algorithms are possible due to the fact that all models within a GWA study differ only by SNP values. In logistic regression, model fitting is done using iterative procedure and exact semi-parallel computations are not possible. We propose an approximate solution exploiting reweighted least-squares technique combined with the fact that SNP effects are very small. In this chapter we also discuss the issues related to the SNP data access. We recommend efficient solutions using available implementations.

In the spirit of matrix operations, we proceed in **Chapter 5** with another fast approach for linear mixed model fitting, called *QuickMix*. The cross-sectional and the longitudinal SNP effects can be estimated with this algorithm. We make a connection between mixed model equations and penalized least squares. We approximate the penalty matrix, so it does not need to be estimated for every model. Many computational tricks are introduced. We avoid operations on large matrices solving the equations in a semi-symbolic way.

In **Chapter 6** we apply both the *QuickMix* and the CTS to the longitudinal BMD data from the Rotterdam Study. Additionally, we discuss many practical issues of large-scale genome scans.

We close with a discussion on our findings and plans for future research in **Chapter 7**.

Chapter 2

Fast linear mixed models for genome-wide association studies with longitudinal data

Abstract

Genome-wide association studies (GWAS) are characterized by an enormous amount of statistical tests performed to discover new disease-related genetic variants (in the form of single nucleotide polymorphisms) in human DNA. Many SNPs have been identified for cross-sectionally measured phenotypes. However, there is a growing interest in genetic determinants of the evolution of traits over time. Dealing with correlated observations from the same individual, advanced statistical techniques need to be applied. The linear mixed model is popular, but also much more computationally demanding than fitting a linear regression model to independent observations. We propose a conditional two-step approach as an approximate method to explore the longitudinal relationship between the trait and the SNP. In a simulation study we compare several fast methods with respect to their accuracy and speed. The conditional two-step approach is applied to relate SNPs to longitudinal BMD responses collected in the Rotterdam Study.

Adapted version of the research article: Sikorska et al.(2013) Fast linear mixed model computations for genome-wide association studies with longitudinal data, *Statistics in Medicine*, 32(1): 165-180

2.1 Introduction

The main aim of genome-wide association studies (GWAS) is to identify common genetic factors across the whole human genome (single nucleotide polymorphisms, SNPs) associated with a particular trait or a disease. In this hypothesis-free approach the entire genome is scanned instead of focusing only on a candidate gene region. Thanks to recent imputation techniques not only directly genotyped but also untyped SNPs can be tested for their association with a trait. The genotype imputation is used to increase the power of the studies and to facilitate performance of a meta-analysis (Li et al., 2009). However, it also increases the number of tests to perform, up to 2.5 mln in Northern Europeans or 40 mln with the advent of the 1000 Genomes project. GWAS are typically set up to detect very small effects ($<$0.5% variance explained) requiring large sample sizes. Much progress has been made over the last decade in the characterization of human genetic variations related to complex diseases. According to the GWAS catalog (Hindorff et al., 2010) 1449 GWAS for 237 traits have been published by June 2011. Dedicated software, e.g. Plink (Purcell et al., 2007), GenABEL (Aulchenko et al., 2007), ProABEL (Aulchenko et al., 2010), has been developed to handle large data files, together with implementations to reduce computation time (Estrada et al., 2009).

Up to now the majority of GWAS has focused on cross-sectionally measured phenotypes. However, much of the epidemiological research is based upon longitudinal designs involving repeated measurements of an outcome of interest gathered for each individual in the study. It may be desirable to identify SNPs that are associated with the longitudinal development of a trait over time. However, repeated measures tend to be correlated and dedicated statistical techniques are needed. Kerner et al. (2009) give a summary of approaches for tackling problems involving longitudinal data coming from the Framingham Heart Study. Several methods have been applied on this data set to characterize the trajectory of a phenotype in association with genetic variants. For a continuous response that evolves over time, the linear mixed model (Laird and Ware, 1982) is a popular approach. Unfortunately this technique can be computationally demanding when many subjects are involved and becomes prohibitively time consuming when it has to be executed a large number of times. A longitudinal GWA study (for 600 000 tests) of cardiovascular disease risk factors in the Bogalusa Heart Study (Smith et al., 2010) analyzed using the linear mixed model took $\approx$3 hours on a cluster of 64 processors. Moreover, in order to fit the final linear mixed model many modeling aspects have to be taken into account, like correct specification of the mean and the variance-covariance structures. To obtain the final results the whole procedure is therefore often repeated several times.

Introduced by Liang and Zeger (1986), Generalized Estimating Equations (GEE) is an alternative approach for modeling correlated data structure, especially if one is interested in the marginal expectations of the outcome as a function of the predictors. GEE models do not specify the joint distribution of a subject's observations, but assume only that the mean and the variance adhere GLM specifications. Intracluster dependency is taken into account via a working correlation matrix. For statistical inference the sandwich estimator of the standard errors is used to correct for possible misspecification of this correlation matrix. GEE models are computationally simpler than the random effects models, however fitting a

few million of them, still remains an issue. The original GEE method requires the data to be missing completely at random (MCAR) since the approach is not likelihood based. Robins et al. (1994) proposed a modification of the standard GEE, so called weighted GEE, which allows the data to be MAR . In their approach, weights corresponding to the inverse probability of missingness are estimated through a logistic regression model, to be later included in the GEE. However, this additional step will even more increase the computational time and for that reason, weighted GEE will not be considered further in this article.

In a GWA analysis with a longitudinal design, where the main interest lies in dependence of the evolution of the outcome over time on the SNP alleles, the p-value of the SNP $\times$ time interaction is of central importance. We argue that the effort needed to evaluate the effects of 2.5 mln SNPs on the longitudinally measured trait is prohibitively large with a classical mixed effects or a GEE approach. It is desirable to develop a fast technique.

We take the classical linear mixed model as the reference. We investigate several fast methods, which approximate the p-value obtained from testing the longitudinal association of the trait with a SNP. Via a simulation study we explore the accuracy and computational times of the fast approaches.

The paper has the following structure. In Section 2.2 we introduce the motivating data set and the research goals. The methods are presented in Section 2.3. In Section 2.4 we describe the setup and the results of a simulation study that compares the performance of the above described approaches. The analysis of the motivating data set is reported in Section 2.5 and final conclusions are stated in Section 2.6.

2.2 Motivating example

Osteoporosis is a skeletal disorder characterized by loss of bone strength and increased risk of fracture. Low bone mineral density (BMD) is one of the most important risk factors for osteoporotic fractures. Mitchell and Yerges-Armstrong (2011) give a summary of the genetic determinants of BMD loss documented by now. Most studies published on the genetics of BMD are based on a one-point measure of this trait. For example, Rivadeneira et al. (2009) provide an elaborate description of 20 BMD loci identified in a large meta-analysis of GWAS. However, such studies cannot provide any information about genetic associations with the rate of bone loss. While the knowledge of the genetic contribution to the baseline variation of BMD grows, no genes have been identified yet as associated with age-related bone changes.

In the prospective population-based Rotterdam Study (Hofman et al., 2013), femoral neck bone mineral density of 4987 elderly individuals (aged 55 years and over) was measured 4 times: at the baseline and after 2, 6, and 12 years. The first three BMD measurements were obtained using a Lunar DPX-L densitometer, while the last measurement was obtained with a Lunar Prodigy densitometer. For the purpose of the calibration, 102 individuals were measured using both devices. We fitted the following cross-calibration linear regression model

$$\mathrm{BMD}_{(\mathrm{DPX-L})} = 1.012\mathrm{BMD}_{(\mathrm{Prodigy})} - 0.019.$$

However, the slope was not significantly different from 1 (95% CI = (0.94, 1.17), $p = 0.74$)

and the intercept was not different from 0 (95% CI = (-0.08, 0.04), p = 0.51). Summary statistics as well as the number of individuals in conjunction with number of non-missing outcome measures can be found in Table 2.1 and Table 2.2. A random sample of individual profiles is shown in Figure 2.1. The exploratory analysis showed that the data are highly unbalanced. The research goal is to identify SNPs associated with evolution of BMD over time. Because of its popularity, a linear mixed model was chosen for the analysis. As typical for GWA studies, the number of considered predictors is much larger than the number of subjects. In such high-dimensional data scenario, multiple regression models break down, so the goal is to fit the linear mixed model for each SNP separately. The chosen model is:

$$\mathrm{BMD}_{ij} = \beta_0 + \beta_1 a_i + \beta_2 w_{ij} + \beta_3 t_{ij} + \beta_4 \mathrm{SNP}_i + \beta_5 t_{ij} \mathrm{SNP}_i + b_{i0} + b_{1i} t_{ij} + \epsilon_{ij},$$

where a_i denotes the age of an individual i at the beginning of the study and w_{ij} describes the weight of individual i at the time t_{ij}. Both covariates are known to be associated with BMD level (Jones et al., 1994; Hannan et al., 2000). Assuming an additive genetic model for imputed data, the SNP variable, which indicates the number of minor alleles in a particular single nucleotide polymorphism, can take values from the interval from 0 to 2. We allow the intercept and the slope for the time evolution to vary from one patient to another, by specifying random effects b_{0i} and b_{1i}. We limit ourselves to a linear evolution of BMD over time. The patients were measured at maximally 4 occasions and exploratory analysis did not indicate any curvilinear pattern. Also, the clinical literature indicates that the age-related decrease in BMD is best described by a linear function (Aloia et al., 1990; Riggs et al., 1982). Adding a quadratic term did not improve the fit and while it resulted in a statistically significant (due to the large sample size) estimated effect, its size appeared to be negligible. The GWA analysis was planned separately for males and females with imputed genotype data (for ≈2.5 mln of SNPs) in R software (version 2.10.0). Computation time for the whole scan (in parallel for two genders) was estimated to be ≈130 days using the R package **nlme** and ≈28 days using the R package **lme4**, on a single desktop with Intel(R) Core(TM) 2 Duo CPU, 3.00GHz.

Table 2.1: Rotterdam study: Evolution of BMD (g/cm^2) measurements over time for females and males.

	Females			Males		
Time	N	mean	SD	N	mean	SD
0	2814	0.829	0.135	2119	0.917	0.137
2	1916	0.827	0.135	1514	0.921	0.136
6	1336	0.822	0.137	1036	0.930	0.140
12	1210	0.818	0.127	982	0.919	0.136

2.3 Statistical methods

Linear mixed model

Let Y_{ij} denote the response variable for the i-th ($i = 1, ..., N$) individual on the j-th ($j = 1, ..., k$) measurement occasion. A vector of all measurements taken on individual

Table 2.2: Rotterdam study: number of individuals with K non-missing responses.

K	Females	Males
4	679	554
3	833	659
2	759	552
1	543	354

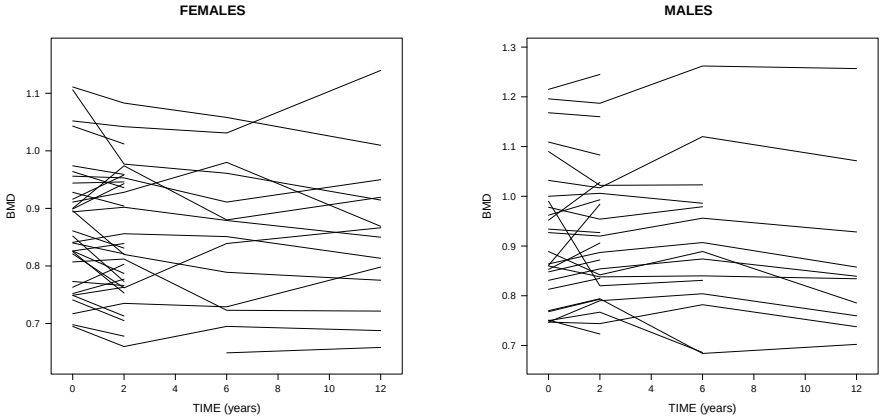


Figure 2.1: Rotterdam Study. Individual profiles of BMD for 40 randomly selected females and males.

i is denoted by Y_i. The linear mixed model is given by :

$$Y_i = X_i\beta + Z_i b_i + \epsilon_i, \quad i = 1, ..., N, \tag{2.1}$$

where X_i, Z_i are $(n_i \times p)$ and $(n_i \times q)$ design matrices, β and b_i are p- and q-dimensional vectors of unknown parameters and ϵ_i is a $(n_i \times 1)$ vector of random errors with $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$. The β parameters are considered as fixed, while the parameters b_i are subject-specific related to the population heterogeneity. We assume $b_i = (b_{0i}, b_{1i})$, where $b_{0i} \sim \mathcal{N}(0, \sigma_0^2)$, $b_{1i} \sim \mathcal{N}(0, \sigma_1^2)$ and $\text{cor}(b_{0i}, b_{1i}) = \rho$. Additionally $b_1, ..., b_N, \epsilon_1, ..., \epsilon_N$ are assumed to be mutually independent.

Inspired by the motivating example (but omitting additional covariates) we assume the following model:

$$Y_{ij} = \beta_0 + \beta_1 S_i + \beta_2 t_{ij} + \beta_3 S_i t_{ij} + b_{0i} + b_{1i} t_{ij} + \epsilon_{ij}, \quad j = 1, ...n_i, \quad i = 1, ..., N, \tag{2.2}$$

where S_i denotes the SNP genotype for an individual i, $S_i \in (0; 2)$ and t_{ij} is the j-th time at which an individual i is measured. We test if there is a statistically significant effect of the SNP on the evolution over time of the response Y_{ij}. This is verified by testing $H_0 : \beta_3 = 0$ in model (2.2) using a Wald, score or likelihood ratio test.

Missing data in longitudinal studies

Although designed to measure every individual at all planned occasions, most of the longitudinal studies result in highly unbalanced data sets. The validity of the methods used for the analysis is then impacted by the missing data mechanism. Little and Rubin (1987) distinguish three mechanisms generating missingness. In the missing completely at random (MCAR) mechanism the probability of observing the response is unrelated to the observed and the unobserved outcome values. On the other hand, if the data are missing at random (MAR), the probability of observing the response depends on the observed outcome values but is independent from the unobserved outcome values. Finally, in the missing not at random (MNAR) mechanism, probability of observing the response depends on observed as well as unobserved outcome values.

We focus on the MCAR and MAR mechanisms. Note that maximum likelihood estimation of β is valid under a MCAR or MAR, as long as the joint distribution of the responses is correctly specified. This implies that estimates from model (2.2) are robust against a MAR process. On the other hand, least squares estimation is robust only under a MCAR process.

Conditional linear mixed model

As shown in Verbeke et al. (2001), misspecification of the cross-sectional component of the linear mixed model can affect the estimation of its longitudinal elements. They propose the so-called conditional linear mixed model which estimates the longitudinal effects regardless of model assumptions on the baseline characteristics.

We rewrite model (2.1) as

$$Y_i = X_i^{(1)}\beta^{(1)} + X_i^{(2)}\beta^{(2)} + Z_i^{(1)}b_i^{(1)} + Z_i^{(2)}b_i^{(2)} + \epsilon_i, \quad i = 1, ..., N, \tag{2.3}$$

where $X_i^{(1)}$ represents the $(n_i \times p_1)$ matrix of time-stationary covariates and $X_i^{(2)}$ $(n_i \times p_2)$ the matrix of time-varying covariates with $X_i = (X_i^{(1)}|X_i^{(2)})$. Additionally $Z_i^{(1)} = \mathbf{1_{n_i}}$, and $Z_i^{(2)}$ is the $n_i \times (q-1)$ matrix of time-varying covariates for the random effects. The conditional linear mixed model uses the approach of conditional inference where random intercepts $b_i^{(1)}$ (b_{i0} in model (2.2)) are treated as nuisance. Estimation is then done conditional on sufficient statistics for the nuisance parameters. Here, in particular $\bar{Y}_i = \sum_j Y_{ij}/n_i$ represents the sufficient statistics for the subject specific intercepts. Verbeke and others showed that inference conditional on $\bar{Y}_i$ is equivalent to inference based on transformed data $A_i^T Y_i$, where A_i is a full-rank $n_i \times (n_i - 1)$ matrix such that $A_i^T \mathbf{1_{n_i}} = 0$.

In the conditional linear mixed model, both sides of the model (2.3) are now multiplied by A_i^T leading to the following model

$$\begin{aligned} Y_i^* \equiv A_i^T Y_i = A_i^T X_i^{(2)}\beta^{(2)} + A_i^T Z_i^{(2)} b_i^{(2)} + A_i^T \epsilon_i = X_i^* \beta^{(2)} + Z_i^* b_i^{(2)} + \epsilon_i^*, \\ i = 1, ..., N. \end{aligned} \tag{2.4}$$

In addition, A_i is chosen such that $A_i^T A_i = I_{n_i - 1}$, so ϵ^* is normally distributed with mean 0 and covariance matrix $\sigma^2 I_{n_i - 1}$. Matrix A_i can be found for each individual with at least two available response measurements. An example of such a matrix is based on orthogonal polynomials (as given in the SAS macro in Verbeke et al. (2001)).

Rewriting $X_i^{(1)}$ into $\mathbf{1_{n_i}} x_i^T$ (x_i is the p_1-dimensional vector of the baseline covariates) we observe that not only the random intercepts but also the cross-sectional fixed effects $\beta^{(1)}$ have vanished from model (2.3). Note that the vector of the n_i original measurements for the i-th individual is reduced to the vector of $n_i - 1$ transformed observations. The transformed response values as well as the transformed design matrix X_i^* are difficult to interpret. However, it has been shown in Verbeke et al. (2001) that the parameter estimates and their standard errors from the conditional linear mixed model are very similar to those obtained from the linear mixed model with correctly specified baseline structure. The conditional linear mixed model corresponding to the model (2.2) is given by:

$$Y_{ij}^* = \beta_2 t_{ij}^* + \beta_3 S_i t_{ij}^* + b_{i1} t_{ij}^* + \epsilon_{ij}^*, \quad j = 1, ..., n_i, \quad i = 1, ..., N, \tag{2.5}$$

since it can be easily shown that $(S_i t_{ij})^* = S_i t_{ij}^*$.

Estimation of the model parameters of the conditional linear mixed model is likelihood-based and therefore it is robust against a MAR process. The additional advantage of the conditional model is that one does not have to specify the cross-sectional part. This model has also another desirable property. Since the longitudinal effects are estimated regardless any baseline characteristics, this approach is also robust against misspecification of the distributional assumptions for the subject specific intercepts.

Generalized Estimating Equations

If one is interested in estimating marginal (population-average) effects and not in individual predictions, GEE is a proper approach (Liang and Zeger, 1986; Diggle et al., 2002). In this semiparametric method, only the first two moments are specified, instead of the full joint distribution. Correlation between the observations is handled via a working correlation matrix. Some popular choices include: independence, exchangeable (compound symmetry), unstructured and autoregressive. There are no strict guidelines for the choice of the correlation matrix. However, if one deals with a small set of unequally spaced time points common for all individuals in the large sample, the unstructured correlation matrix is one of the most suitable choices. Those 'working' assumptions about the correlation structure are corrected by computing the sandwich estimator for the standard errors. Fitting a GEE model is computationally simpler than fitting a random effects model. However, it still requires a lot of effort if it has to be repeated many times. Note also that since GEE models are not likelihood-based, they are not robust against the MAR process. Statistical inference is done using a Wald or a score test.

In our case the marginal model is:

$$E(Y_{ij}) = \beta_0 + \beta_1 S_i + \beta_2 t_{ij} + \beta_3 S_i t_{ij}, \quad j = 1, ..., n_i, \quad i = 1, ..., N. \tag{2.6}$$

Fast approaches

Because of its many desirable properties, a linear mixed model is our favorite method to make an inference about the longitudinal association between the considered outcome and the SNPs. The p-value for SNP $\times$ time effect in model (2.2) is of our main interest. We

explored three fast alternative methods that can provide approximately the same p-value for testing $H_0 : \beta_3 = 0$ in model (2.2), in a much shorter time. These methods split the analysis into two steps in order to avoid fitting the full model (2.2) repeatedly for every SNP. The first step, reduces the vector of n_i observations for each individual, to at most 2 outcomes (intercept and slope of linear evolution). However, the data reduction depends on the method. In the second step one regresses the individual slopes on each of the SNPs. Note that in classical linear regression the likelihood ratio test, the Wald test and the score tests coincide.

Slope as outcome

In the first stage, the slope $\beta_{1i}^{\triangle}$ per individual (with at least two available observations) is estimated from the model

$$Y_{ij} = \beta_{0i}^{\triangle} + \beta_{1i}^{\triangle} t_{ij} + \epsilon_{ij}^{\triangle} \quad j = 1, ..., n_i \quad i = 1, ..., N, \tag{2.7}$$

using a least squares approach. In the second stage the estimated $\hat{\beta}_{1i}^{\triangle}$'s are regressed on S_i by fitting

$$\hat{\beta}_{1i}^{\triangle} = \beta_0^{\triangle\triangle} + \beta_1^{\triangle\triangle} S_i + \epsilon_i^{\triangle\triangle}, \quad i = 1, ..., N, \tag{2.8}$$

using ordinary least squares. Notice that this approach is based to the two-stage formulation of the linear mixed model. The p-values of interest are those obtained from testing $H_0 : \beta_1^{\triangle\triangle} = 0$ and they are compared to the p-values for SNP$\times$time interaction term in model (2.2).

According to the theory described in Section 2.3, the slope as outcome approach is not robust against the MAR mechanism since the estimation in the first step is based on the least squares principle.

Two-step

In the first step all terms containing S_i are omitted from model (2.2), so it becomes

$$Y_{ij} = \beta_0^* + \beta_1^* t_{ij} + b_{0i}^* + b_{1i}^* t_{ij} + \epsilon_{ij}^* \quad j = 1, ..., n_i \quad i = 1, ..., N. \tag{2.9}$$

The linear mixed model (2.9) is then fitted only once. In case the SNP is important (cross-sectionally or longitudinally), model (2.9) is misspecified. In that case we expect that the subject-specific slopes predicted from this model will contain information about differences of the evolution of Y_{ij} over time between SNP alleles (S_i). Therefore, in the second step we regress the best linear unbiased predictors (BLUPs, Henderson (1975)) of b_{1i}^* on S_i with a simple linear regression model:

$$\hat{b}_{1i}^* = \beta_0^{**} + \beta_1^{**} S_i + \epsilon_i^{**} \quad i = 1, ..., N. \tag{2.10}$$

We argue that testing $H_0 : \beta_1^{**} = 0$ in model (2.10) results in approximately the same p-value as that obtained from testing $H_0 : \beta_3 = 0$ in model (2.2).

Assuming that β_1 and β_3 in model (2.2) are 0, the reduced model (2.9) (without SNP-terms included) is still the correct model describing evolution of the response Y_{ij}. Because the estimation is performed using maximum likelihood, the two-step approach is robust against MCAR and MAR mechanisms.

Conditional two-step

This method is similar to the previous one, however, in the first step we use the idea of the conditional inference. We apply the data transformation introduced in Section 2.3, but like in the two-step approach, omitting SNP from the model. Now the reduced model becomes independent from all cross-sectional effects. Namely, the model is as follows.

$$Y_{ij}^{*} = \beta_2^{**} t_{ij}^{*} + b_{i1}^{**} t_{ij}^{*} + \epsilon_{ij}^{**}, \quad j = 1, ..., n_i, \quad i = 1, ..., N. \tag{2.11}$$

Model (2.11) is still misspecified by the lacking SNP×time interaction term. We again argue that the subject specific slopes from the reduced conditional linear mixed model (2.11) will contain an information about omitted SNP. Note also that the SNP data were not transformed and the original values are taken to the second step. Step 2 is then based on estimating the regression coefficients in model

$$\hat{b}_{1i}^{**} = \beta_0^{***} + \beta_1^{***} S_i + \epsilon_i^{***}, \quad i = 1, ..., N, \tag{2.12}$$

using ordinary least squares. The conditional two-step approach, is again robust against the MCAR and MAR processes for the same reasons as the two-step approach.

2.4 Simulation study

Setup

Throughout the simulation study we take the mixed effects model as our reference. We focus on the p-values obtained from testing the longitudinal relationship of the outcome with the SNPs. As faster alternatives to the full modeling we explore the three approaches proposed in Section 2.3. We also applied the classical GEE approach for two choices for the working correlation matrix: unstructured and independence. The first choice appears to be most suitable for our data structure. The latter is expected to be the least computationally demanding. Except from the GEE approach, all above methods are considered as approximate to the linear mixed model. The objective of the simulation study is two-fold. First, we assessed the accuracy of the approximation using a measure $\mathrm{SD}_{\mathrm{diff}}$ defined as a standard deviation of $\log_{10}(p_F) - \log_{10}(p_A)$, where p_F and p_A are the p-values for testing $\beta_3 = 0$ from model (2.2) and alternative approaches, respectively. We also explored the probability of type I error and the power of the surrogates relative to the linear mixed model (2.2). We also compare computational times necessary to conduct a GWA analysis for approximately 2.5 mln SNPs. The data were generated to resemble as much as possible the motivating example. Namely, the data sets were simulated according to the model (2.2) for all individuals measured at 4 fixed occasions ($t_{ij} = 0, 2, 6, 12$). The fixed effects and the

variance-covariance matrix were set similar to those coming from fitting the reduced linear mixed model (excluding all terms containing SNP) for BMD data for females and can be found in Table 2.3. In addition, we explore the dependence of the approximation on the sample size, unknown SNP effects and the process generating missing responses. For that reason we designed a full factorial design with 4 following factors:

1. sample size (with 3 levels: N=500, 1000, 3000)
2. cross-sectional SNP effect (with 2 levels: $\beta_1 = 0$ or $\beta_1 = 0.005$)
3. longitudinal SNP effect (with 2 levels: $\beta_3 = 0$ or $\beta_3 = 0.0008$)
4. missing data process (with 3 levels: complete data, MCAR dropout and MAR dropout)

In the MCAR dropout, at each time point we randomly select individuals that are lost to follow-up. The missingness is generated such that the percentage of patients with available response at each of 4 time points is set to 100, 70, 50, 30% of the sample size. The MAR dropout was generated according to the logistic model

$$\log\left(\frac{p_{ij}}{1-p_{ij}}\right) = 1.5 - 2.5Y_{ij-1}, \quad j = 2, 3, 4,$$

where p_{ij} is the probability of missing response for the subject i at the time point j. This dropout model generates the proportions of available outcomes which are practically the same as for MCAR scenario.

We simulated 200 data sets for each of the 36 combinations of the factorial design and analyzed them using the linear mixed model (LMM) and five alternative methods:

1. slope as outcome (SAO),
2. two-step (TS),
3. conditional two-step (CTS),
4. GEE-unstructured (GEE-un),
5. GEE-independence (GEE-ind),

The simulation study was conducted in R software (version 2.10.0 R Core Team (2013)). The linear mixed model was fitted using the R package **lme4** and the GEE approach was performed with the R packages **gee** and **geepack**. The function *lmer* (in **lme4**) does not provide the p-values. Based on estimated parameters and their standard errors, we calculated them using a Wald test. This test (based on robust standard errors) was also used for the GEE models. The parameters were estimated using REML. The results were also compared to the ML estimation. The conclusions did not differ between the two methods.

Results

Graphical and numerical comparisons of the statistical efficiency

We explored the influence of several factors on the performance of the alternative approaches. We now discuss some general patterns found in the simulation results. For all

Table 2.3: Values of the parameters in model (2.2) used in the simulation study

Effect	Parameter	Value
Intercept	β_0	0.970
Time effect	β_2	-0.004
sd(b_0)	σ_0	0.110
sd(b_1)	σ_1	0.003
cor(b_0, b_1)	ρ	-0.140
sd(ϵ)	σ	0.040

the scenarios we calculated the proposed measure of accuracy, $\mathrm{SD}_{\mathrm{diff}}$. The results for the MCAR and MAR dropouts were very similar, therefore we focus on the MAR dropout results. For a balanced design, as readily seen from Figures 2.2 and 2.3, the p-values from all the alternative methods are very close to those obtained in the LMM. The calculated $\mathrm{SD}_{\mathrm{diff}}$'s are given in Tables 2.4 and 2.5. We observe that in case of a non-zero longitudinal SNP effect, the $\mathrm{SD}_{\mathrm{diff}}$'s are larger than for data sets simulated under the null. This is due to the fact that our measure of accuracy is based on $-\log_{10}(p)$ which penalizes more the deviations on the tails of the z-score distribution. We also notice that the performance of the two-step approach is affected by a cross-sectional SNP effect. As theoretically motivated in Section 2.3, the conditional two-step is robust against misspecifying the baseline characteristic. Moreover, for the balanced design the inferences from the slope as outcome and the conditional two-step approaches are identical.

Different conclusions emerge for unbalanced designs. Here, the approximations of the p-values can be very inaccurate for all the methods except from the conditional two-step approach. We observe a lack of precision for the slope as outcome and the GEE approaches (Figure 2.4). The calculated $\mathrm{SD}_{\mathrm{diff}}$'s are summarized in Tables 2.6 and 2.7. Similarly to balanced designs, the performance of the two-step approach is influenced by the cross-sectional SNP effect (Figure 2.5). The conditional two-step is again robust against this factor. If there is a true nonzero longitudinal SNP effect, the slope as outcome and the GEE methods tend to overestimate the p-values with respect to those from the LMM (Figure 2.6). The same tendency is observed for the two-step approach in case of a nonzero cross-sectional SNP effect (Figure 2.7).

In Tables 2.8 and 2.9 we show the calculated type I error rates. The conclusions for the balanced and unbalanced scenarios appear to be the same. The type I error rates are similar for all the methods and do not differ distinctly between the LMM and the approximate approaches. With respect to the power, the conclusions diverge between balanced and unbalanced designs (Tables 2.10 and 2.11). Unlike the complete data scenarios, where the power of the alternative methods is similar to that of the LMM, in case of dropouts in the study some approaches are highly underpowered. Namely, as already mentioned above the highest overestimation of the p-values occurs for SAO and GEE-ind. The power of CTS is very close to the power of LMM and also higher than for the two-step approach.

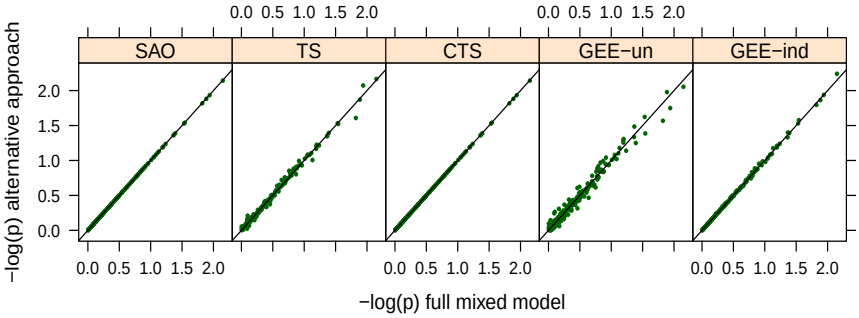


Figure 2.2: Simulation study ($\beta_1 = 0$ and $\beta_3 = 0$, N=1000), balanced scenario. On the x-axis $-\log_{10}(p)$ for SNP×time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the approximate methods (SAO - slope as outcome, TS - two-step, CTS - conditional two-step, GEE-un - GEE with unstructured working correlation matrix, GEE-ind - GEE with independent working correlation matrix)

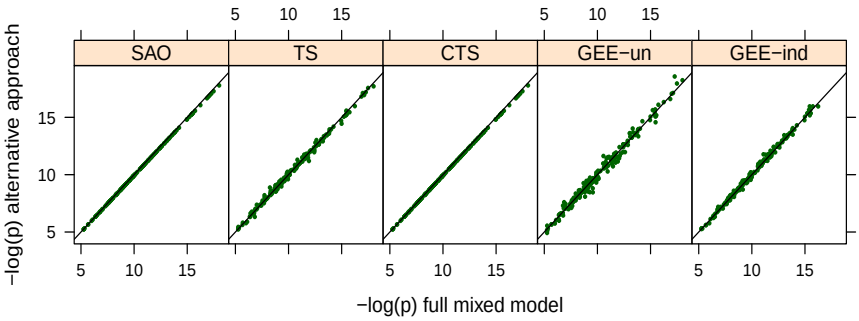


Figure 2.3: Simulation study ($\beta_1 = 0$ and $\beta_3 = 0.008$, N=3000), balanced scenario. On the x-axis $-\log_{10}(p)$ for SNP×time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the approximate methods (SAO - slope as outcome, TS - two-step, CTS - conditional two-step, GEE-un - GEE with unstructured working correlation matrix, GEE-ind - GEE with independent working correlation matrix)

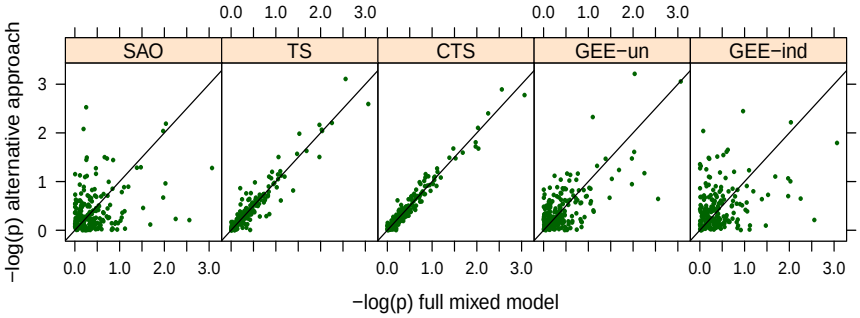


Figure 2.4: Simulation study ($\beta_1 = 0$ and $\beta_3 = 0$, N=1000), MAR scenario. On the x-axis $-\log_{10}(p)$ for SNP×time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the approximate methods (SAO - slope as outcome, TS - two-step, CTS - conditional two-step, GEE-un - GEE with unstructured working correlation matrix, GEE-ind - GEE with independent working correlation matrix)

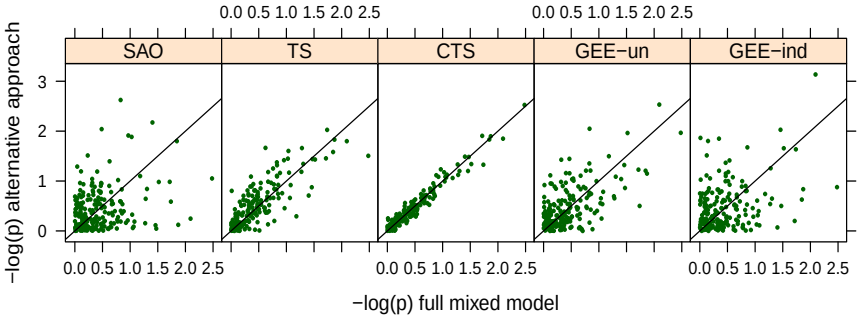


Figure 2.5: Simulation study ($\beta_1 = 0.005$ and $\beta_3 = 0$, N=3000), MAR scenario. On the x-axis $-\log_{10}(p)$ for SNP×time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the approximate methods (SAO - slope as outcome, TS - two-step, CTS - conditional two-step, GEE-un - GEE with unstructured working correlation matrix, GEE-ind - GEE with independent working correlation matrix)

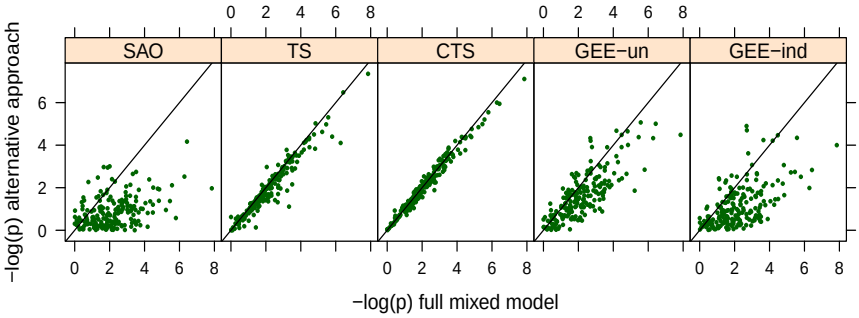


Figure 2.6: Simulation study ($\beta_1 = 0$ and $\beta_3 = 0.008$, N=1000), MAR scenario. On the x-axis $-\log_{10}(p)$ for SNP×time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the approximate methods (SAO - slope as outcome, TS - two-step, CTS - conditional two-step, GEE-un - GEE with unstructured working correlation matrix, GEE-ind - GEE with independent working correlation matrix)

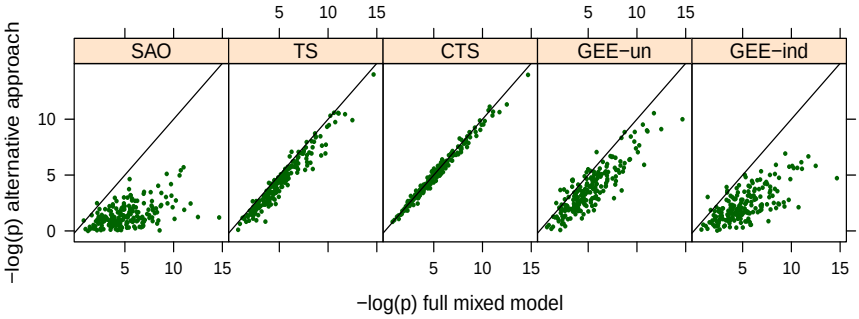


Figure 2.7: Simulation study ($\beta_1 = 0.005$ and $\beta_3 = 0.008$, N=3000), MAR scenario. On the x-axis $-\log_{10}(p)$ for SNP×time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the approximate methods (SAO - slope as outcome, TS - two-step, CTS - conditional two-step, GEE-un - GEE with unstructured working correlation matrix, GEE-ind - GEE with independent working correlation matrix)

Table 2.4: Simulated data, balanced scenario. $\beta_3 = 0$

	SD_{diff}					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
SAO	0.003	0.001	0.0004	0.002	0.001	0.0006
TS	0.047	0.033	0.0240	0.063	0.059	0.0452
CTS	0.003	0.001	0.0004	0.002	0.001	0.0006
GEE-un	0.082	0.061	0.0489	0.087	0.067	0.0442
GEE-in	0.019	0.013	0.0082	0.018	0.014	0.0078

Table 2.5: Simulated data, balanced scenario. $\beta_3 = 0.0008$

	SD_{diff}					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
SAO	0.03	0.04	0.04	0.03	0.03	0.04
TS	0.11	0.13	0.18	0.18	0.24	0.30
CTS	0.03	0.04	0.04	0.03	0.03	0.04
GEE-un	0.23	0.23	0.33	0.20	0.24	0.30
GEE-in	0.10	0.13	0.29	0.11	0.14	0.33

Table 2.6: Simulated data, MAR dropout. $\beta_3 = 0$

	SD_{diff}					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
SAO	0.59	0.55	0.51	0.59	0.57	0.52
TS	0.17	0.16	0.14	0.21	0.22	0.27
CTS	0.11	0.11	0.09	0.13	0.09	0.09
GEE-un	0.37	0.38	0.40	0.42	0.40	0.37
GEE-in	0.56	0.54	0.49	0.53	0.51	0.51

Table 2.7: Simulated data, MAR dropout. $\beta_3 = 0.0008$

	SD_{diff}					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
SAO	1.00	1.23	1.90	0.96	1.21	2.07
TS	0.29	0.38	0.43	0.40	0.52	0.60
CTS	0.16	0.22	0.32	0.16	0.22	0.32
GEE-un	0.64	0.81	1.18	0.62	0.80	1.16
GEE-in	0.87	1.11	1.71	0.80	1.12	1.73

Table 2.8: Simulated data, balanced scenario. $\beta_3 = 0$

	P(type I error)					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
LMM	0.04	0.05	0.05	0.06	0.05	0.03
SAO	0.04	0.05	0.05	0.06	0.05	0.03
TS	0.04	0.05	0.05	0.05	0.05	0.03
CTS	0.04	0.05	0.05	0.06	0.05	0.03
GEE-un	0.05	0.05	0.05	0.07	0.06	0.03
GEE-in	0.04	0.05	0.05	0.05	0.05	0.03

Table 2.9: Simulated data, MAR dropout. $\beta_3 = 0$

	P(type I error)					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
LMM	0.06	0.06	0.05	0.07	0.07	0.06
SAO	0.05	0.05	0.05	0.05	0.05	0.04
TS	0.05	0.06	0.04	0.07	0.06	0.08
CTS	0.06	0.06	0.05	0.06	0.07	0.05
GEE-un	0.08	0.04	0.04	0.08	0.05	0.05
GEE-in	0.07	0.06	0.05	0.06	0.05	0.06

Table 2.10: Simulated data, balanced scenario. $\beta_3 = 0.0008$

	Power(%)					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
LMM	82	95	100	78	97	100
SAO	82	95	100	78	97	100
TS	82	95	100	78	97	100
CTS	82	95	100	78	97	100
GEE-un	80	95	100	78	97	100
GEE-in	82	95	100	77	97	100

Table 2.11: Simulated data, MAR dropout. $\beta_3 = 0.0008$

	Power(%)					
	$\beta_1 = 0$			$\beta_1 = 0.005$		
Method	N=500	N=1000	N=3000	N=500	N=1000	N=3000
LMM	38	74	98	44	66	99
SAO	12	23	46	14	26	44
TS	34	69	96	36	56	95
CTS	36	72	99	42	69	98
GEE-un	32	56	93	32	50	91
GEE-in	16	36	75	22	32	75

Computation times

The approximate system times needed to fit 2.5 mln of models (in R software) on a desktop (with Intel(R) Core(TM) 2 Duo CPU, 3.00GHz) for different sample sizes are provided in Table 2.12. It is clear that necessity of fitting large number of random effects models causes computational issues. We observe a high impact of sample size on the computational times, which may become very large. A full GWA analysis for a sample of 3000 individuals even if performed using the fastest available program (the R package **lme4**) would take more than a month. As expected, the GEE methods are less time demanding, but still much slower than the two-step approaches. Simplification of the working correlation matrix from unstructured to independence almost halves the time. We conclude that much time can be gained using considered in this paper fast approaches. All of them are about 170 times faster (for sample size of 3000) than the linear mixed model approach. It is also important to note that all the reported times are based on the model without any additional covariates. Thus, they are very optimistic. Parallelization across multiprocessor systems and/or using GRID computing approaches (Estrada et al., 2009) may further reduce the time achieving reasonable time frames.

Table 2.12: Simulation study. Approximate system time (R function) for 2.5 mln models in R software (Intel(R) Core(TM) 2 Duo CPU, 3.00GHz)

	System Time (h)		
Method	N=500	N=1000	N=3000
LMM (lme)	660	1250	3500
LMM ($lmer$)	180	347	833
GEE-un (gee)	90	139	430
GEE-ind ($geepack$)	35	55	139
SAO (lm)	3	4	5
TS (lm)	3	4	5
CTS (lm)	3	4	5

2.5 Analysis of the BMD data

Based on the simulation study, we conclude that the conditional two-step approach is the most accurate from the fast approaches. It was chosen for the analysis of the motivating data set. The data transformation necessary to remove the cross-sectional terms was done using a SAS macro provided by Verbeke et al. (2001). Next, for the transformed data, we fitted a linear mixed model with only slope specified in the random part. As shown in Section 2.2, the BMD data set is highly unbalanced. The predicted subject-specific slopes for the individuals with incomplete profiles are then shrunken toward the population-average mean profile. That resulted in a distribution of the random slopes which is symmetric, but with a high kurtosis. However, the same feature of the predicted individual slopes was observed in the simulation study, in case of generated dropouts, where the true distribution of subject specific intercepts and slopes was multivariate normal. Predicted BLUPs for the random slope were used as a response for a simple linear regression. This last step was

performed using MACH2QTL (Li et al., 2010) as implemented in the web-based interface GRIMP.

Under the hypothesis of no association between SNPs and the trait the calculated p-values follow a uniform(0,1) distribution (Sellke et al., 2001). It is common to summarize a GWA analysis by Q-Q and Manhattan plots. In the Q-Q plot the distribution of the calculated $-\log_{10}(p)$'s is compared to the theoretical distribution under the null (such as χ^2). The confidence limits are estimated based on the fact that j-th order statistics from a uniform(0,1) sample has a beta$(j, n-j+1)$ distribution . In Manhattan plots the calculated $-\log_{10}(p)$'s are plotted against the SNPs positions on the chromosomes. The resulting Q-Q and Manhattan plots for the BMD data for females are shown in the Figures 2.8 and 2.9. No SNPs reached the genome-wide significance level ($p < 5 \times 10^{-8}$, Pe'er et al. (2008); Frazer et al. (2007)). Similarly, no SNPs reaching that level of significance were found for males (plots not shown).

Finally, 1000 randomly selected SNPs from the GWA analysis for females (using the conditional two-step approach) were also analyzed with the full linear mixed model approach. The resulting plot of the corresponding p-values (Figure 2.10) confirmed that the method can be used as a LMM surrogate. The $\mathrm{SD}_{\mathrm{diff}}$ for that sample of SNPs was calculated as 0.117.

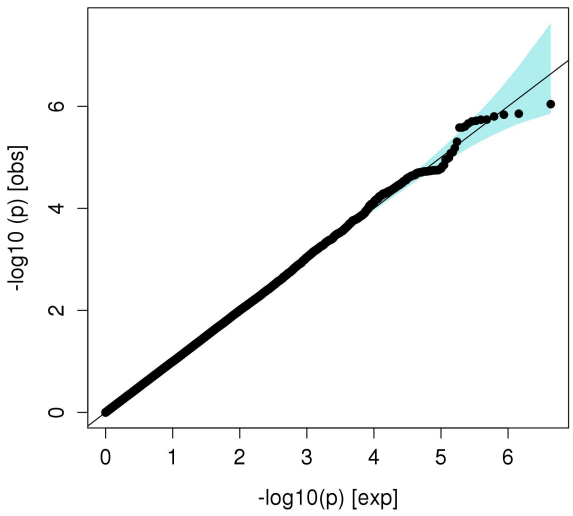


Figure 2.8: Rotterdam Study, females. Q-Q plot of GWA analysis using CTS approach. The gray band represents pointwise 95% confidence interval

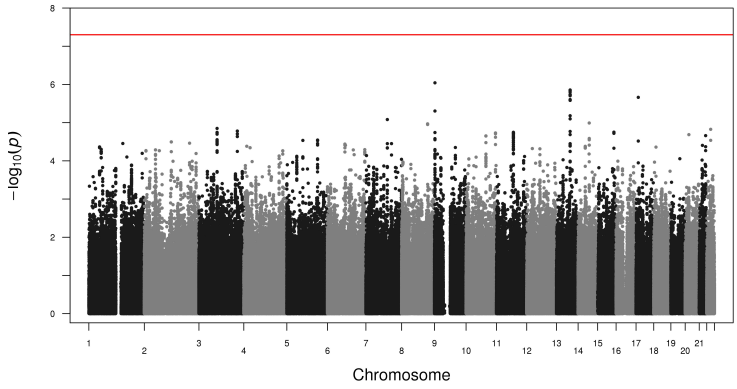


Figure 2.9: Rotterdam Study, females. Manhattan plot of GWA analysis using CTS approach. The horizontal line represents the genome-wide significance level

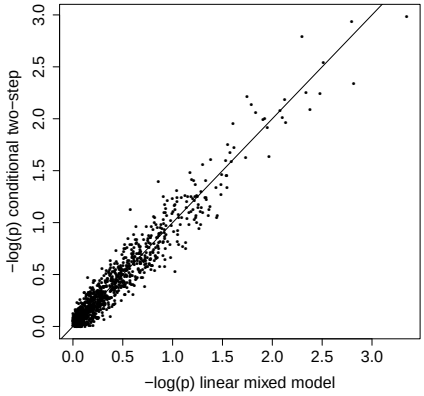


Figure 2.10: BMD data, females. On the x-axis $-\log_{10}(p)$ for SNP $\times$ time term from the linear mixed model and on the y-axis the corresponding $-\log_{10}(p)$ from the conditional two-step

2.6 Conclusions

In this article we discuss computational issues in genome-wide association studies in case of a longitudinal design with a normal response. We explored several approximate methods which can analyze the longitudinal relationship of the response with the SNP in a much

shorter time. The simulation study showed that the best performing method is the conditional two-step. Among other considered techniques, the CTS combines two desirable properties: the highest accuracy and the shortest CPU time. It also showed some additional advantages comparing to the two-step. Performance of the CTS is independent from the cross-sectional effect of the SNPs. The practical application of the conditional two-step approach is to perform the first genome-wide scan for millions of SNPs in a fast manner, to later on focus on the promising hits via the full linear mixed model. Of next interest is to explore further potential of the conditional two-step. Several data scenarios can be considered, for example an unbalanced design with missing responses generated by missing not at random (MNAR) process. We also find it worthy to examine the properties of the CTS in case of more complicated data structure like non-normal or correlated errors.

Chapter 3

GWAS with longitudinal phenotypes - performance of approximate procedures

Abstract

In our previous work (Sikorska et al., 2013b) we proposed the conditional two-step (CTS) approach as a fast method providing an approximation to the p-value for the longitudinal SNP effect. In the first step a reduced conditional linear mixed model is fitted, omitting all the SNP terms. In the second step the estimated random slopes are regressed on SNPs. The CTS has been applied to the bone mineral density data from the Rotterdam Study and proved to work very well even in unbalanced situations. In another article (Sikorska et al., 2013a) we suggested semi-parallel computations, greatly speeding up fitting many linear regressions. Here, we explore further the properties of the CTS approach both analytically and by simulations. We also investigate speedup achieved by combining the CTS approach with semi-parallel regression.

Adapted version of the research article: Sikorska et al.(2014) GWAS with longitudinal phenotypes - performance of approximate procedures. *European Journal of Human Genetics.* Under revision.

3.1 Introduction

In longitudinal studies repeated measurements from the same participant are gathered over a period of time. Such studies play an important role in clinical and epidemiological research since they can relate changes in an individual to covariates. Longitudinal studies have been recently introduced in genome-wide association studies, where the goal is to find single nucleotide polymorphisms (SNPs) which impact change in physical condition of individuals. Hundreds of diseases and traits have been investigated cross-sectionally, identifying thousands of significant SNPs. For several traits it is relevant to explore their change over time. In this article the evolution of bone mineral density (BMD) in elderly people participating in the Rotterdam Study (Hofman et al., 2013) is taken as a guiding example.

Measurements taken from the same individual are correlated which invalidates the basic assumption of a linear regression model. Therefore dedicated statistical procedures are required. In practice participants are often not examined at regular time points, they stop the study permanently (dropouts) or miss visits (intermittent missingness). The linear mixed model (LMM) is one of the popular approaches to analyze such irregularly measured responses. Fitting one LMM to thousands of individuals takes around a second. However, performing the LMM computations millions of times makes the whole-genome scans prohibitive in practice, especially with the growing amount of SNP data implied by the 1000 Genomes Project. Additionally, the model building process may require repetition of the whole analysis for different mean and/or covariance structures.

Mixed models have been intensively used in GWA studies involving related individuals where the dependence structure needs to be properly modeled. This is also time consuming. Speeding up mixed models in this context is therefore also important and received quite some attention, see e.g. Lippert et al. (2011) and Zhou and Stephens (2012). However, limited research has been devoted in this respect for longitudinal data.

It is expected that only a few SNPs correlate with the change of the trait over time. The longitudinal effect of a SNP is measured by the SNP$\times$time interaction term in the mean structure of the model. Current GWAS are interested in identifying markers for which the p-value is lower than the threshold of 5×10^{-8}. Sikorska et al. (2013b) explored several approximate procedures which identify the important SNPs in a fast manner. In particular, the authors proposed the conditional two-step approach which is based on the conditional linear mixed model (Verbeke et al., 2001). They explored the properties of this method on longitudinal bone mineral density data collected in the Rotterdam Study and compared their proposal to several other approaches. The conditional two-step proved to be an excellent approximation to the LMM approach. The CTS approach is basically reducing the computations to fitting one linear mixed model in the first step and in the second step a simple regression model, for each SNP at a time.

Sikorska et al. (2013a) showed how to achieve huge speedups in the second step via so called semi-parallel regression (SPR). Many SNPs are analyzed at the same time using big matrix operations, which replace time consuming loops. In this way, a GWAS with simple linear regression is performed 50-60 times faster than with standard implementations. Solutions for efficient SNP data access have also been discussed in Sikorska et al. (2013a). As

a result, the combination of the CTS and SPR, makes an analysis of a GWAS with longitudinal data feasible even on a desktop computer, thereby considerably reducing demands on computing resources.

Here we further investigate the properties of the conditional two-step approach, analytically as well as by simulations. In addition, we compare it to a related method, the two-step approach. The latter approach has already been applied in some GWAS. Our goal is to explore the robustness of the two approximate methods for different data scenarios allowing us to draw general conclusions. Moreover, we discuss the speed gains achieved by applying jointly the CTS and SPR. Our simulations lead us to the discussion on the practical aspects of the fast analysis of a longitudinal GWAS. Finally, in the Supplementary Material we provide R code useful for the implementation of the CTS approach.

3.2 Materials and methods

The development of fast approximate procedures was inspired by the data collected in the Rotterdam Study, exploring determinants of disease and disability in Dutch individuals age 55 years and over. In this prospective cohort study the bone mineral density of more than 5000 individuals, aged 55 or over, was measured at baseline (in 1990) and after approximately 2, 6, and 12 years. After an extensive whole-genome research on the cross-sectional BMD (Rivadeneira et al., 2009) it was decided to explore genetic contributions to the change of BMD over time in elderly people. The BMD data from the Rotterdam Study are unbalanced and the missingness rates at the second, third and fourth recording times were 30, 50 and 70%, respectively. Due to the unbalanced structure of the data, the linear mixed model was chosen for the analysis. Originally the model was corrected for the age at entry to the study and the evolution of body weight. However, for ease of illustration, in this article we consider only time and SNP. Below, we indicate how additional covariates should be handled in practice. In the Supplementary Material we provide simulations proving that conclusions remain the same when other covariates are included in the model.

The linear mixed model

The linear mixed model describing the vector y_i, which consists of n_i measurements taken on individual i over time, can be expressed as

$$y_i = X_i\beta + Z_i b_i + \epsilon_i, \quad i = 1, ..., N \tag{3.1}$$

where X_i and Z_i are $n_i \times p$ and $n_i \times q$ design matrices for fixed and random effects. The fixed effects model the overall population characteristic and are common to all individuals with the same X_i. The random effects describe the individual deviation from the average population evolution. Additionally ϵ_i represents a $n_i \times 1$ vector of measurement errors. We adopt model (3.1) to our motivating example, describing the response BMD for an individual i at the occasion j as follows:

$$y_{ij} = \beta_0 + \beta_1 s_i + \beta_2 t_{ij} + \beta_3 s_i t_{ij} + b_{0i} + b_{1i} t_{ij} + \epsilon_{ij}. \tag{3.2}$$

In model (3.2) the fixed effects are represented by SNP (s_i), time (t_{ij}) and their interaction ($s_i t_{ij}$). We assume that the response evolves over time in a linear fashion. The SNP variable can be represented either by an integer from {0,1,2} denoting the genotyped number of the reference allele or a continuous number between 0 and 2 describing the expected genotype count after imputation. The subject-specific part of the model consists of a random intercept (b_{0i}) and a random slope (b_{1i}). The first describes an individual deviation of the baseline BMD level from β_0. The latter characterizes the subject-specific fluctuation of the slope around $\beta_2 + \beta_3 s_i$. Classically, it is assumed that the random effects have a bivariate normal distribution with mean 0 and covariance matrix

$$D = \begin{bmatrix} \sigma_0^2 & \rho\sigma_0\sigma_1 \\ \rho\sigma_0\sigma_1 & \sigma_1^2 \end{bmatrix}.$$

Finally, ϵ_{ij} denotes a normally distributed measurement error with mean 0 and variance σ^2. It is assumed that the ϵ_{ij} are independent from $b_i = (b_{0i}, b_{1i})^T$. From the above it follows that the n_i-dimensional response y_i has covariance matrix given by $V_i = Z_i D Z_i^T + \sigma^2 I_{n_i}$, where Z_i is a $n_i \times 2$ dimensional matrix with ones in the first column and t_{ij} in the second columns and I_{n_i} is the identity matrix of size n_i. More information about the mixed model formulation can be found in Verbeke and Molenberghs (2009).

The fixed effects and the unknown variance components in (3.2) are commonly estimated iteratively using (restricted) maximum likelihood ((RE)ML). The parameter estimates (apart from the SNP terms) of model (3.2) for the BMD data applied to women are shown in Table 3.1.

Table 3.1: Estimates of the parameters in model (3.2) from the BMD data for women obtained from a LMM analysis of the Rotterdam Study (taken from Sikorska et al. (2013b)).

Effect	Parameter	Estimate
Intercept	β_0	0.970
Time effect	β_1	-0.004
sd(b_0)	σ_0	0.110
sd(b_1)	σ_1	0.003
cor(b_0, b_1)	ρ	-0.140
sd(ϵ)	σ	0.040

Our main interest lies however in the estimate of β_3 and more precisely in the p-value for testing $H_0 : \beta_3 = 0$. In the Appendix we show that, when the data are balanced (so $t_{ij} = t_j$ and $n_i = n$), the ML estimate of β_3 is equal to:

$$\widehat{\beta}_3 = \frac{\text{cov}(\mathbf{s}, \mathbf{u}) - \bar{\mathbf{t}}\,\text{cov}(\mathbf{s}, \mathbf{y})}{n\,\text{var}(\mathbf{t})\text{var}(\mathbf{s})}, \tag{3.3}$$

where $\mathbf{t} = (t_1, ..., t_{n_i})^T$, $\bar{\mathbf{t}} = \sum_j t_j/n$, $\text{var}(\mathbf{t}) = \sum_j (t_j - \bar{t})^2/n$, $\mathbf{s} = (s_1, ..., s_N)^T$, $\text{var}(\mathbf{s}) = \sum_i (s_i - \bar{s})^2/N$, $\mathbf{y} = (y_1, ..., y_N)^T (y_i = \sum_j y_{ij})$, $\mathbf{u} = (u_1, ..., u_N)^T (u_i = \sum_j y_{ij} t_j)$. Assuming that the variance components are known, the variance of $\widehat{\beta}_3$ is given by (see

Appendix):

$$\text{var}(\widehat{\beta}_3) = \frac{\sigma^2 + \sigma_1^2\, n\, \text{var}(\mathbf{t})}{N\, n\, \text{var}(\mathbf{t})\text{var}(\mathbf{s})}. \tag{3.4}$$

In practice the unknown σ^2, σ_0^2 and σ_1^2 are replaced by their (RE)ML estimates. The ratio $\widehat{\beta_3}/\widehat{SE(\beta_3)}$ gives the value of the t-statistic which determines p^*, the p-value for the SNP$\times$time effect.

The conditional linear mixed model (CLMM)

The conditional linear mixed model has been suggested when baseline characteristics are not of interest or cannot be properly modeled. Verbeke et al. (2001) and Verbeke and Fieuws (2007) showed that misspecification of the cross-sectional part of the model may lead to biased estimates of the longitudinal part. Such a misspecification happens when, e.g. an important cross-sectional SNP effect has been omitted from model (3.2). We are interested only in the longitudinal part of the model and therefore we aim for unbiasedly estimating the longitudinal part irrespective of estimating the cross-sectional part. Below we explain why that is particularly useful. The idea behind the CLMM is to map the time-stationary part of the model to zero. This is achieved by multiplying both sides of the model (3.1) by a full-rank $n_i \times (n_i - 1)$ matrix A_i such that $A_i^T \mathbf{1}_{\mathbf{n_i}} = 0$ and $A_i^T A_i = I_{(n_i-1)}$, where $\mathbf{1}_{\mathbf{n_i}}$ is a n_i-length vector of ones. In our case, the CLMM corresponding to (3.2) has the following form:

$$y_{ij}^* = \beta_2 t_{ij}^* + \beta_3 s_i t_{ij}^* + b_{1i} t_{ij}^* + \epsilon_{ij}^*, \tag{3.5}$$

where $y_{ij}^* = A_i^T y_{ij}$, $t_{ij}^* = A_i^T t_{ij}$, $\epsilon_{ij}^* = A_i^T \epsilon_{ij}$, $\text{var}(b_{1i}^*) = \sigma_1^2$ and $\text{var}(\epsilon_{ij}^*) = \sigma^2$.

Matrix A_i can be easily found using properties of orthogonal polynomials. One should note that this data transformation reduces the length of the vector of observations for individual i from n_i to $n_i - 1$. Thus, only individuals with at least two repeated measurements actually contribute to the estimation of the CLMM parameters. Although the transformed variables are not directly interpretable, we are interested in the relationships of the estimated fixed and random effects between the linear mixed model and its conditional version. If the linear mixed model is correctly specified, the estimates from the LMM and the CLMM are the same in the balanced case (Verbeke et al., 2001). In the unbalanced case, the empirical evidence suggest that they are similar (Verbeke et al., 2001). But other operational characteristics, such as the type I error rate and the power of the CLMM versus the LMM have not yet been investigated, to our best knowledge. When the cross-sectional part of the LMM is wrong, the CLMM may prevent bias in estimation of the longitudinal effects.

The data transformation is easily done in SAS, as shown in Verbeke et al. (2001) or in R using the code provided in the Supplementary Material.

Approximate procedures

Fitting model (3.2) for one SNP takes around 4 seconds in the R package **nlme** (Pinheiro et al., 2013) and around 1 second in the package **lme4** (Bates et al., 2014). In a GWAS millions of such models may need to be fitted, which results in weeks or months of computations on a single computer. Even when applying the commonly used brute-force approach

of multiprocessor computing, the computation time may be prohibitively long. It is desirable to simplify the computations making GWA scans for longitudinal data quicker and feasible even on one desktop computer. Below we show how a GWA analysis based on a mixed model can be reduced to fitting simple linear regression models to each SNP providing an approximate p-value for the hypothesis test $H_0 : \beta_3 = 0$ for each SNP separately.

Two-step (TS)

We can reformulate model (3.2) into

$$y_{ij} = \beta_0^* + \beta_2^* t_{ij} + b_{0i}^* + b_{1i}^* t_{ij} + \epsilon_{ij}, \tag{3.6}$$

with

$$\beta_0^* = \beta_0 + \beta_1 E(s),$$

$$\beta_2^* = \beta_2 + \beta_3 E(s),$$

$$b_{0i}^* = b_{0i} + \beta_1 (s_i - E(s)),$$

$$b_{1i}^* = b_{1i} + \beta_3 (s_i - E(s)).$$

We call model (3.6) the reduced model and its fitting constitutes the first step in the two-step procedure. The procedure consists in practice in fitting a linear mixed model which omits both SNP terms from model (3.2). Therefore, it has to be done only once for all the SNPs. All additional covariates (time-stationary and time-varying) should be also included in the reduced model. Note that b_{0i}^* and b_{1i}^* have still zero means satisfying conditions of linear mixed model formulation. However, their variance-covariance matrix, D^* has changed into

$$D^* = \begin{bmatrix} \sigma_0^2 + \beta_1^2 \text{var}(s) & \rho\sigma_0\sigma_1 + \beta_1\beta_3 \text{var}(s) \\ \rho\sigma_0\sigma_1 + \beta_1\beta_3 \text{var}(s) & \sigma_1^2 + \beta_3^2 \text{var}(s) \end{bmatrix}. \tag{3.7}$$

The variance of the b_{1i}^* suggests that if a SNP has an effect on the longitudinal evolution of the trait, information about that effect is now hidden in the individual slopes. The unknown values of $b_i^* = (b_{0i}^*, b_{1i}^*)^T$ are estimated using empirical Bayesian methods resulting in so-called "best linear unbiased predictors" (BLUPs, Robinson (1991)). The estimation is done according to the following equation:

$$\widehat{b_i^*} = D^* Z_i^T W_i^* (y_i - X_i^* \beta^*), \tag{3.8}$$

where X_i^* and β^* are the design matrix and the vector of fixed effects of model (3.6). The matrix W_i^* is equal to V_i^{*-1}, where $V_i^* = Z_i D^* Z_i^T + \sigma^2 I_{n_i}$. In practice, β^* and D^* are replaced by their (RE)ML estimates. In our analytical derivations, we assume that the variance of the measurement error will not change much from that in model (3.2).

The second step of the two-step approach involves regressing the estimated random slope $\widehat{b_{1i}^*}$ on the omitted SNP using the following simple regression model

$$\widehat{b_{1i}^*} = \beta_0^{**} + \beta_1^{**} s_i + \epsilon_i^{**}. \tag{3.9}$$

We are interested in the relationship of the p-value from testing the hypothesis $H_0 : \beta_1^{**} = 0$ with p^*. It can be shown (see Appendix) that the ML estimate of β_1^{**} in the balanced case has the form

$$\widehat{\beta}_1^{**} = \frac{\text{cov}(\mathbf{s}, \mathbf{u}) - n\, c\, \bar{\mathbf{t}}\, \text{cov}(\mathbf{s}, \mathbf{y})}{\text{var}(\mathbf{s})(\sigma^2/\sigma_1^{*2} + \sum_j t_j^2 - c(\sum_j t_j)^2)}, \tag{3.10}$$

where σ_0^{*2} and σ_1^{*2} are the diagonal elements of D^* and $c = (n + \sigma^2/\sigma_0^{*2})^{-1}$. It can be shown using elementary algebraic manipulations that $|\widehat{\beta}_1^{**}| \leq |\widehat{\beta}_3|$. This illustrates the shrinkage effect of BLUP estimators, see e.g. Robinson (1991).

We note that expression (3.10) is based on the assumption that the covariance part in D^* is zero. If there is no cross-sectional effect of the SNP ($\beta_1 = 0$), this implies that ρ must be zero. One can always turn the covariance of the original random intercept and slope into zero by choosing an appropriate translation of the time. That is, when t_{ij} is replaced by $t_{ij} - a$ with $a = -\rho\, \sigma_0/\sigma_1$ the covariance of the changed random effects becomes zero. However, such a change in origin drastically changes the other settings of the model, e.g. the time variable does not start anymore from zero. Consequently, we cannot compare the transformed situation with the situation whereby the correlation is zero at the start. When β_1 is not zero, the covariance will be equal to $\beta_1 \beta_3 \text{var}(s)$. However, its value will be relatively small, because of the very small SNP effects in a GWAS setting.

To see how (3.3) and (3.10) are related, one takes σ^2 much smaller than σ_0^{*2} and σ_1^{*2}. Then $c \approx n^{-1}$ and $\widehat{\beta}_1^{**} \approx \widehat{\beta}_3$. However, in many practical situations this assumption will not hold. For the standard error of $\widehat{\beta}_1^{**}$ no insightful expression could be obtained. Therefore the relationship between p^* and the p-value for $H_0 : \beta_1^{**} = 0$ remains unclear and needs to be evaluated numerically.

Conditional two-step (CTS)

The conditional linear mixed model corresponding to model (3.2) is given by (3.5). The transformed outcome y_{ij}^* is a function of only longitudinal effects including the effect of interest: SNP×time interaction. Following the rationale from the two-step approach we build a reduced conditional linear mixed model

$$y_{ij}^* = \beta_2^{\triangle} t_{ij}^* + b_{1i}^{\triangle} t_{ij}^* + \epsilon_{ij}^{\triangle}, \tag{3.11}$$

with $\text{var}(b_{1i}^{\triangle}) = \sigma_1^{*2}$ and as in the two-step approach, $\text{var}(\epsilon_{ij}^{\triangle}) \approx \sigma^2$. Note that all additional baseline covariates vanish from the conditional linear mixed model through the data transformation. However, the transformed time-varying covariates remain in the reduced CLMM. The idea is now to regress the EBLUPs of $b_{1i}^{\triangle}$ on SNPs via the following simple regression model

$$\widehat{b}_{1i}^{\triangle} = \beta_0^{\triangle\triangle} + \beta_1^{\triangle\triangle} s_i + \epsilon_i^{\triangle\triangle}. \tag{3.12}$$

The ML estimate for the SNP effect in model (3.12) and its variance are for the balanced case given by

$$\widehat{\beta}_1^{\triangle\triangle} = \frac{\text{cov}(\mathbf{s}, \mathbf{u}) - \bar{\mathbf{t}}\, \text{cov}(\mathbf{s}, \mathbf{y})}{\text{var}(\mathbf{s})(n\, \text{var}(\mathbf{t}) + \sigma^2/\sigma_1^{*2})}, \tag{3.13}$$

$$\text{var}(\widehat{\beta}_1^{\triangle\triangle}) = \frac{n\, \text{var}(\mathbf{t}) \sigma_1^{*2}}{N\, \text{var}(\mathbf{s})(\sigma^2/\sigma_1^{*2} + n\, \text{var}(\mathbf{t}))}. \tag{3.14}$$

It is easy to relate $\widehat{\beta}_3$ and $\widehat{\beta}_1^{\triangle\triangle}$ by

$$\widehat{\beta}_1^{\triangle\triangle} = \frac{n \text{ var}(\mathbf{t})}{n \text{ var}(\mathbf{t}) + \sigma^2/\sigma_1^{*2}} \widehat{\beta}_3. \tag{3.15}$$

Now the shrinkage effect of the BLUPs is immediately clear from the above relationship. For the Rotterdam study, this shrinkage factor is about 0.32. The relationship between the t-statistics is given by:

$$t_{CTS} = \sqrt{\frac{n \, \sigma_1^2 \text{var}(\mathbf{t}) + \sigma^2}{n \, \sigma_1^{*2} \text{var}(\mathbf{t}) + \sigma^2}} \, t_{LMM}, \tag{3.16}$$

where t_{CTS} and t_{LMM} are the t-statistics for the CTS approach and the LMM, respectively. Since also the variance of $\widehat{\beta}_1^{\triangle\triangle}$ is shrunken compared to that of $\widehat{\beta}_3$, the t-statistic of the CTS approach is not necessarily smaller in absolute value. In GWAS, SNP effects are usually very small, which means that $\sigma_1^{*2} \approx \sigma_1^2$. Consequently, $t_{CTS} \approx t_{LMM}$, implying approximately the same p-values for the two methods.

Note that we have compared the performance of the CTS to LMM, while it is in fact an approximation to the CLMM. The LMM was chosen as comparator, because of its popularity. But, for the reasons stated above we might have chosen also the CLMM to compare with, because for the balanced case $t_{CLMM} = t_{LMM}$ (Verbeke et al., 2001; Verbeke and Fieuws, 2007).

While our analytical derivations lead to a clear relationship between p^* and the p-value from the conditional two-step approach, for the two-step approach things are less clear. Unknown remains also the impact of cross-sectional SNP effect on the two-step approach. Moreover, our derivations are limited to balanced data with known variance components, which rarely occurs in practice. Performance of the two-step and the conditional two-step approaches in more practical situations is addressed in a simulation study.

Simulation study

The settings in our simulation study are based on the characteristics of the BMD data. In particular we assume that the data for 2000 individuals come from model (3.2) with the parameter values equal to those in Table 3.1. For the balanced scenario, we assumed that the measurements were taken for all individuals at baseline and after 2, 6, and 12 years without missing data. The SNP variable was taken as a random number with a uniform distribution on [0, 2]. We denote this setting as Scenario 1. Then we considered modifications of that scenario whereby the values for ρ and σ_1^2 were changed. The scenarios are described in Table 3.2. We ran an additional eight scenarios which are the same as in Table 3.2 but with $\sigma = 0.04$ (inspired by value in Table 3.1) replaced by $\sigma/2$. But, since the results were quite the same, we did not include these results here.

Table 3.2: Scenarios used in the simulation study for the balanced case describing changes made to the parameters values with respect to those given in Table 3.1. All settings assume $\beta_1 = 0$.

Scenario	ρ	σ_1
1	-0.14	0.003
2	0	0.003
3	0	0.03
4	-0.5	0.003
5	-0.9	0.003
6	0.5	0.003
7	0.9	0.003
8	-0.9	0.03

In the unbalanced case, we assumed that times of measurements after baseline are slightly different between individuals. We used a jittering function which adds the times from the balanced case a random number between -0.8 and 0.8. Additionally we simulated a missing at random (MAR) dropout. The MAR mechanism assumes that the probability of missing observation depends on the observed outcome values but is independent from the unobserved values (Little and Rubin, 1989). More specifically, we assumed that the probability of dropping out from the study at time t_{ij} ($j > 1$) depends on y_{ij-1} according to the following logistic model:

$$\log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \alpha + \beta y_{ij-1}, \tag{3.17}$$

where p_{ij} is the probability of a missing y_{ij}. The values of α and β determine how important the dropout is and were chosen such that the dropout at the second measurements was about 30%, implying a dropout at the third and fourth occasions of around 50% and 70% respectively. The simulation scenarios for the MAR case were chosen the same as for the balanced case (Table 3.2).

To evaluate the two approximate procedures several criteria were considered. Comparison is done mainly with the LMM, for the reasons stated above. First, there is the probability of type I error. Preferably, we would like to have it around $\alpha = 0.05$ as for the LMM. Secondly, the power should be close to the power of the LMM. We also evaluated the precision defined as the standard deviation of $\log_{10}(p_{\text{LMM}}) - \log_{10}(p_{\text{A}})$, where p_{A} is the p-value from the approximating method. We denote this measure as SD_{diff}. Lastly, we are interested in the influence of β_1 on the approximations given by the two-step approach. Two values for β_1 were chosen: 0.01 and 0.05. The first one is the estimate obtained in the cross-sectional GWA analysis for BMD data.

All simulations were performed using R software (R Core Team, 2013). The LMMs were fitted using the package **lme4**. In our experience, this package is faster and encounters less problems with convergence than the package **nlme**.

Probit regression to estimate power

In the mixed model framework the distributions of the test statistics for the Wald, t- and F-tests are generally known only under the null hypothesis (Verbeke and Molenberghs, 2009). Exhaustive simulations are considered the most accurate method to compute statistical power. However fitting many LMMs is time consuming and so is the simulation-based estimation of the power curve. The effect and parameters are computed repetitively for a grid of values. The proportion of times that a SNP is qualified as significant gives the empirical type I error rate (simulation under null hypothesis) and power (simulation under alternative hypothesis). For an exhaustive simulation study this approach demands a discouraging amount of computation time.

We propose a faster way for power calculations based on the probit model. Helms (1992) demonstrated via simulations that the distribution of the general F-test

$$H_0 : \xi \equiv L\beta - \xi_0 = 0 \text{ versus } H_A : \xi \neq 0$$

under the alternative can be approximated by a noncentral F-distribution with noncentrality parameter δ given by

$$\delta = \xi^T [L(\textstyle\sum_i X_i^T V_i^{-1} X_i)^{-1} L^T]^{-1} \xi.$$

From this it follows that the t-test for testing

$$H_0 : \beta_3 = 0 \text{ versus } H_A : \beta_3 = \theta,$$

under H_A has a noncentral t-distribution with noncentrality parameter $\sqrt{\delta}$ (Littell, 2006). In GWAS situations the number of degrees of freedom is large and the (non-central) t-distribution can be approximated by a normal distribution. Consequently, the power curve plotting the effect size versus the statistical power has approximately the shape of the cumulative normal distribution function. We observed the same shape for the approximate procedures. In our simulations, a grid of equally spaced β_3-values is chosen on the interval $[0, \beta_{3,max}]$, where $\beta_{3,max}$ is the smallest value for which the power is practically 100%. Thousand grid values were chosen and for each value one data set was simulated and the considered models were fitted. The obtained p-values can be dichotomized according to the condition $p < 0.05$ giving p_{bin}. Finally, a probit model was fitted to p_{bin}.

Results

Balanced case

The results of the simulation study for the balanced case are summarized in Table 3.3. We observe a high impact of the correlation between random effects (ρ) on the performance of the two-step approach. For the variance of the random slope like in the BMD data, the power of the two-step decreased with about 31% and 91% for ρ equal to -0.5 and -0.9 respectively. When σ_1^2 was further increased, the TS procedure revealed only a minor loss of power (0.05%) even when the correlation ρ was set to -0.9 (Scenario 8, Table 3.3). A positive sign of the correlation affected slightly less the approximation. The conditional two-step approach exhibits a stable behavior across all the scenarios, resulting in a similar

type I error rate and power as LMM. The difference in performance between TS and CTS is also demonstrated in Figure 3.1. The minimal loss of power for CTS in Scenario 8 may be caused by a small difference between σ^2 and σ^{*2} in case of larger β_3 simulated for that scenario.

Unbalanced case

When the data are unbalanced, both approximate methods are sensitive for the changes in ρ and σ_1^2, but the effect on the CTS approach is often minimal. The results from our simulations are shown in Table 3.4. We observed a loss of power for TS and CTS when $|\rho|$ increases. However, TS was more affected by a large $|\rho|$. For $\rho = -0.5$, CTS lost up to 2.4% of power while TS was highly underpowered (max loss of 53%). As for the balanced case, increasing σ_1^2 (Scenario 8) improved the approximation for TS, but did not reduce power loss for the CTS. For all scenarios, the type I error rates for TS and CTS were similar to LMM. The performance of the approximations for the unbalanced case is illustrated in Figure 3.2.

Heteroscedasticity and robust standard errors

The variance of the estimated BLUPs from the LMM is given by

$$\text{var}(\widehat{b_i}) = DZ_i^T \left(W_i - W_i X_i \big(\sum_i X_i^t W_i X_i \big)^{-1} X_i^T W_i \big) \right) Z_i D.$$

From this expression we observe that the variation of the estimated individual slopes may be quite different between individuals, especially in case of unbalanced data, which could lead to heteroscedasticity in the second step of the two-step procedures. However, replacing the simple linear regressions by weighted linear regressions with weights obtained in the first step, had almost no impact on our simulation results. In general, one might of course prefer the weighted regressions solutions at the expense of a small extra computation time.

Effect of distributional assumptions

We explored the performance of the approximate procedures in cases where the distributional assumptions of the LMM are not met. We considered MAR dropout and two modifications of Scenario 1. In the first modification we simulated measurement error from the strongly asymmetric, exponential distribution with rate parameter equal to σ_1 to keep the same variance like in the BMD data. We next shifted this distribution such that the mean was equal to 0. In the second case we applied the exponential distribution to the random effects, also keeping variances like those in the BMD data. For the exponentially distributed measurement error, the type I error rate was approximately 0.05 for all the methods and the maximum loss of power was equal to 7% and 1% for TS and CTS, respectively.

Similarly, for the exponentially distributed random effects, the maximum loss of power for both TS and CTS was only 1% with the type I error rate approximately 0.05. Note that in this case the random effects were simulated as independent.

Table 3.3: Balanced case. Comparison of probability of type I error, loss of power with respect to LMM and precision measured by SD_{diff} of two-step and conditional two-step.

	P(type I error)			Max loss of power		SD_{diff}	
Scenario	LMM	TS	CTS	TS	CTS	TS	CTS
1	0.044	0.045	0.044	0.48%	0.00%	0.05	0.006
2	0.047	0.045	0.047	0.40%	0.00%	0.17	0.006
3	0.059	0.058	0.059	0.23%	0.00%	0.03	0.015
4	0.063	0.052	0.063	31.00%	0.00%	0.66	0.007
5	0.053	0.054	0.053	91.00%	0.00%	1.38	0.007
6	0.061	0.052	0.061	27.00%	0.00%	0.72	0.007
7	0.053	0.062	0.053	67.00%	0.00%	1.19	0.007
8	0.056	0.056	0.056	0.50%	0.21%	0.12	0.015

Table 3.4: Unbalanced case. Comparison of probability of type I error, loss of power with respect to LMM and precision measured by SD_{diff} of two-step and conditional two-step.

	P(type I error)			Max loss of power		SD_{diff}	
Scenario	LMM	TS	CTS	TS	CTS	TS	CTS
1	0.053	0.056	0.053	5.50%	0.20%	0.26	0.15
2	0.029	0.036	0.033	1.20%	1.10%	0.20	0.11
3	0.066	0.067	0.066	0.31%	0.03%	0.08	0.07
4	0.055	0.052	0.055	53.40%	2.40%	1.04	0.30
5	0.055	0.058	0.052	94.30%	11.00%	1.51	0.49
6	0.054	0.050	0.052	43.90%	0.67%	0.90	0.11
7	0.052	0.043	0.058	84.00%	4.30%	1.29	0.25
8	0.047	0.047	0.050	40.10%	13.00%	1.52	0.90

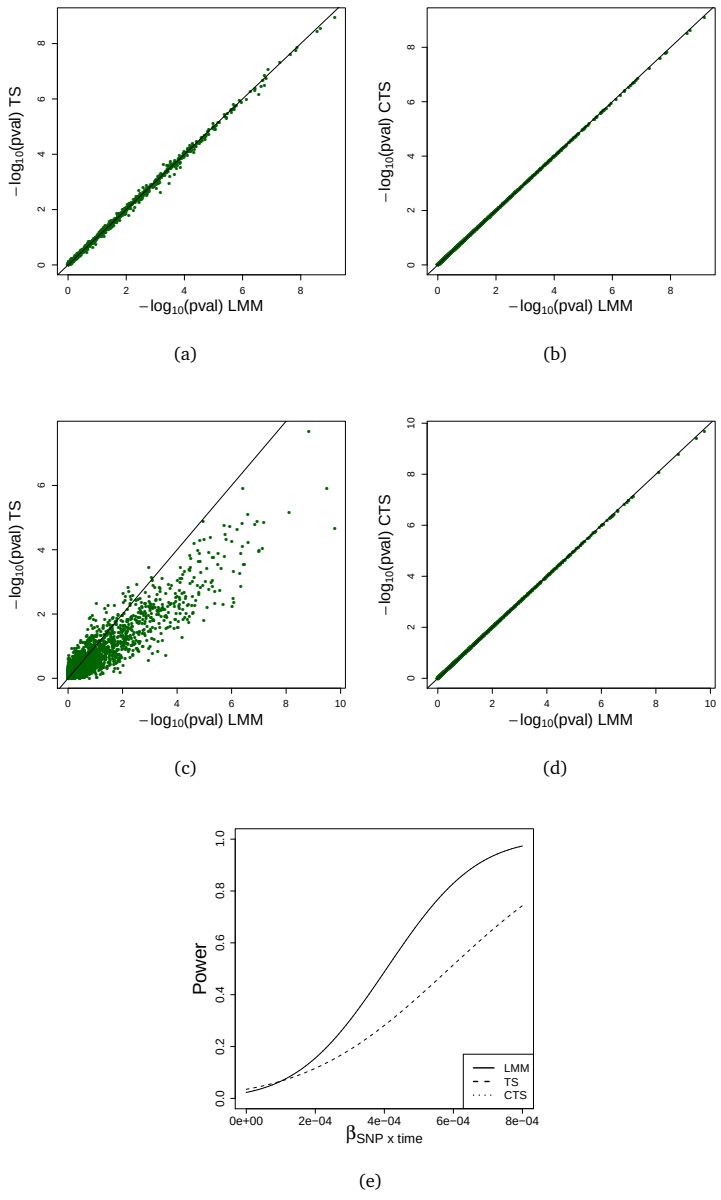


Figure 3.1: Balanced case, Scenarios 1 and 4. Panels (a) and (b) display the approximation of the p-values obtained in TS and CTS for Scenario 1. Panels (c) and (d) display the approximation for Scenario 4. Panel (e) shows the power curves for Scenario 4. The curves for LMM and CTS are overlapping.

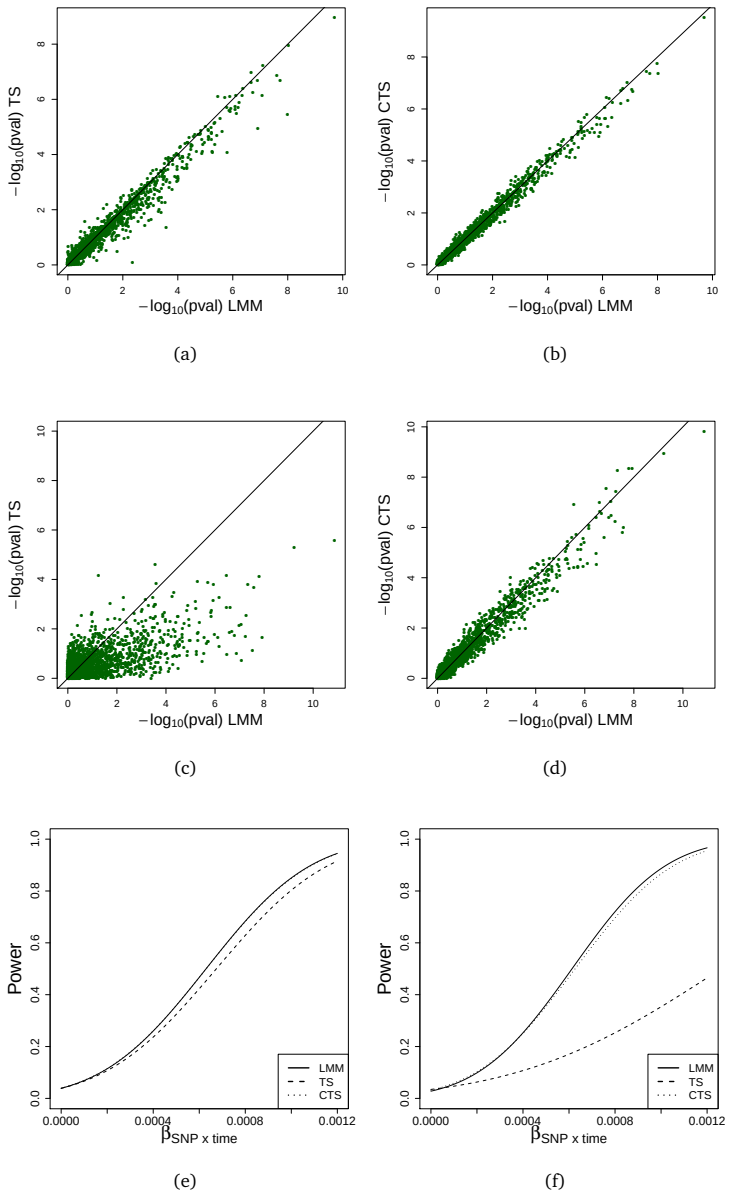
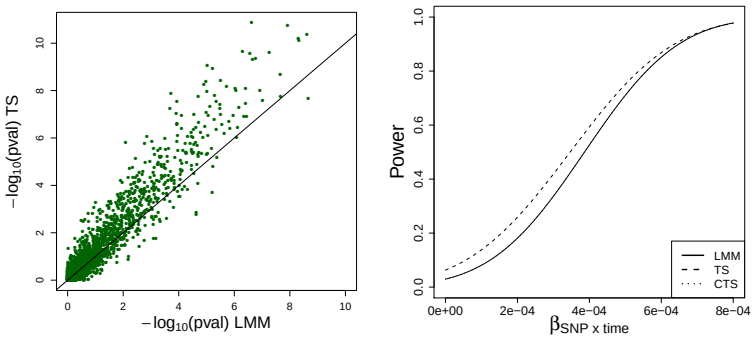


Figure 3.2: MAR case, Scenarios 1 and 4. Panels (a) and (b) display the approximation of the p-values obtained in TS and CTS for Scenario 1. Panels (c) and (d) display the approximation for Scenario 4. Panels (e) and (f) show the power curves for Scenarios 1 and 4.

Influence of the cross-sectional SNP effect

Our simulations showed a big impact of the cross-sectional SNP effect on the performance of TS. In the balanced case, when $\beta_1 = 0.01$ only a minor power loss of the TS was observed (1%). However, for $\beta_1 = 0.05$, the type I error rate was inflated to 0.10 compared to 0.038 for LMM and CTS. The performance of TS for that scenario is displayed in Figure 3.3. For the MAR case, a cross-sectional SNP effect of 0.01 led to a loss of power for the TS to even 17% (Figure 3.4). For $\beta_1 = 0.05$, the type I error rate was inflated to 0.54.



(a) (b)

Figure 3.3: Balanced case, Scenario 1 with $\beta_1 = 0.05$. Panel (a) shows the approximation given by the two-step. Panel (b) shows the power curves for LMM, TS and CTS.

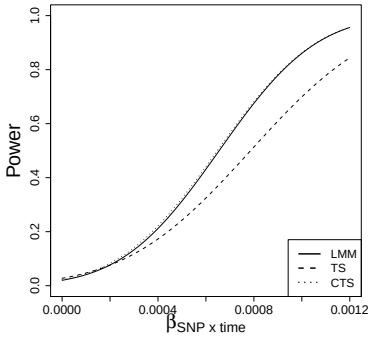


Figure 3.4: MAR case, Scenario 1 with $\beta_1 = 0.01$. Power curves for the LMM, TS and CTS.

Approximation to conditional linear mixed model

The CTS approach provides an approximation to the slope obtained by the CLMM. Now, since the CLMM removes the effect of the cross-sectional part on the estimation of the slope, different results from those of the LMM should be expected. As mentioned above, the advantage of the CLMM is that it is less vulnerable to misspecification of the cross-sectional part. In order to appreciate the CTS as an approximation of the CLMM, we also compared the p-value obtained from the CTS and the CLMM. For this, we only deal with the unbalanced case of Scenario 5, where the greatest drop in power is seen. From Supplementary Figure 1 it is clear that the CTS perfectly approximates the p-value obtained in the CLMM, suggesting that its power loss with respect to LMM is implicitly related to the conditional model.

Discussion of the simulation results

We can safely conclude from the above simulation results that the CTS approach is superior to the TS approach, loosing basically no power in the balanced case compared to the LMM and showing a minimal power loss in the unbalanced case. The eventual loss in power is due to the loss of power with the CLMM, and this loss must be weighted against the advantage of the conditional approach which is less vulnerable to model assumptions than the LMM.

In the balanced case, the TS seriously suffers from lack of power in Scenarios 4-7, where small variability of the random slope is combined with a high correlation between the random effects. However, it is not the high correlation per se that causes the drop in power since one can always render the correlation zero. In fact, it is the complex interplay between σ, σ_0, σ_1, ρ and the values $t_1, \ldots, t_n$ that have an impact on the power. Our simulation study did not unravel this complex interplay, but had only the intention to show that some loss of power can be expected in some extreme situations. Below we show that this does not have an effect on the proposed practical procedure.

Furthermore, we observed that the behavior of the CTS is quite similar to that of the CLMM. That there is some loss in power of the CLMM approach might be surprising given the results in Section 4.2 of Verbeke et al. (2001). In that section it is argued that the CLMM implies no loss of information from a Bayesian viewpoint. However, this result is based on the assumption that the random intercept has a flat prior, while we have taken the classical assumption of joint normality for the random intercept and slope. That in the balanced case there is (basically) no power loss and in the unbalanced case there must be in general a power loss, can be seen from the following reasoning.

The results in Section 4.3 of the same paper applied to the current simplified situation results in:

$$p(\boldsymbol{y} \mid \boldsymbol{b}_0, \beta_3, \boldsymbol{b}_1, \sigma^2) = p(\boldsymbol{y}^* \mid \beta_3, \boldsymbol{b}_1, \sigma^2)\, p(\overline{\boldsymbol{y}} \mid \boldsymbol{b}_0, \beta_3, \boldsymbol{b}_1, \sigma^2),$$

with $\boldsymbol{y}$ the stacked vector of responses, $\boldsymbol{b}_0$ ($\boldsymbol{b}_1$) the stacked vector random intercepts (slope), $\boldsymbol{y}^*$ the stacked vector of $\boldsymbol{y}_i^*$ values and $\overline{\boldsymbol{y}}$ the stacked vector of profile means. Now for each of the profile means the following result holds:

$$\overline{y}_i | b_{0i}, b_{1i} \sim \mathrm{N}(\beta_0 + \beta_1 s_i + \beta_2 \bar{t}_i + \beta_3 s_i \bar{t}_i + b_{0i} + b_{1i}\bar{t}_i, \sigma^2),$$

with $\bar{t}_i$ the average time for the i-th subject. In the balanced case, one can change the time origin such that $\bar{t}_i \equiv \bar{t} = 0$ without changing anything on the estimation of the longitudinal part of the model. This implies that there is no information on β_3 anymore in the second part of the likelihood (part of $\overline{\boldsymbol{y}}$). That a minimal loss of information for the CTS approach was seen in some of the simulations for the balanced case, has to do with the estimation of the variance parameters. Indeed, the variance parameters of the LMM are present in both parts of the likelihood and must therefore be estimated better with the LMM than with any of the two parts separately. However, in the unbalanced case, no change in origin can remove β_3 from that part. Hence, a loss of power is expected with the CLMM and also with the CTS approach, but the loss of power is often minimal as seen in the simulations.

Practical aspects and computation times

Our simulation study clearly indicates that CTS should be the method of choice for the approximate computations in longitudinal GWAS. We illustrated that especially for unbalanced data (and we expect the readers to be mainly dealing with such data) this approach is more precise and reliable than the TS approach. In our experience, CTS does not inflate the type I error and leads to only a minor loss of power, which depends on the data scenario. In practice, one can learn a lot about the data by fitting a LMM without a SNP. The variance components parameters will basically not change when the SNP is included in the model, as the effects in GWAS are very small. This provides a useful information on the expected power loss when applying the CTS approach. Additionally, is it advisable to conduct a small simulation study, say for 100 SNPs, assessing the quality of approximations. After the approximate GWAS is performed, one will obviously confirm the findings by fitting the LMM to the most promising SNPs. Depending on the expected small power loss, a somewhat increased number of "the top" SNPs can be considered. The practical use of the conditional two-step approach is also displayed in Supplementary Figure 2.

We compared the pure computation times (without time spent on data access) using the 3.0.2 64-bit version of R software (R Core Team, 2013) on a desktop computer with i5-3470, 3.20 GHz and 8 GB of RAM. Fitting one linear mixed model (3.2) for 2000 individuals takes around 2 seconds using the package **nlme** and around 0.5 seconds using the package **lme4**, which would imply 23 and 6 days respectively, for 1M of SNPs. This computation time is linear in the sample size. Applying the two two-step procedures we reduce the computations to fitting one LMM for all the SNPs and a simple linear regression for each SNP at a time. One should also note that the data transformation needed for CTS is performed within a minute. Using a standard procedure in R, function *lm*, fitting 1 million regressions takes around 1 hour. Applying the semi-parallel regression we can perform this analysis within 2 minutes. Finally, one should add the time spent on re-analyzing "the top" SNPs with the LMM, which is around 2 minutes for fitting 100 regression models. To summarize, we speed up the computations for 1M of SNPs from respectively 23 or 6 days to 5 minutes.

Supplementary Figure 3 displays the time needed to perform the second step of the two-step procedures and the speeding up with respect to the linear mixed model for different sample sizes and the number of repeated observations. Note that the computation

time for the two-step methods is essentially the same regardless of the number of longitudinal observations. With 10 observations per individual the conditional two-step approach is 50000 times faster than the function *lmer*. This is partially due to the fact that the values for a SNP, which are read from the files as N-dimensional vector, do not need to be expanded to length $N * n$, which is an additional cost of the standard analysis in R functions.

Another aspect of the analysis is SNP data access, which remains the same issue for any type of computations. Application of array-oriented binary files has been discussed in Sikorska et al. (2013a) showing that an additional 5 minutes are needed to access the data for 1M of SNPs. As a result, we make the GWAS computations feasible on a single everyday computer.

3.3 Discussion

We explored the performance of two approximate procedures for GWAS on longitudinal data in different scenarios. Our analytical investigations for the balanced case showed that the conditional two-step approach provides an excellent approximation of the p-value for the SNP$\times$time interaction term obtained from the LMM and the CLMM. This result was also confirmed by the simulation study. The performance of the two-step approach is less straightforward, due to the lack of insightful expression for the standard errors. For the balanced case this method showed to be sensitive for the variance-covariance structure of the random effects. The same behavior was observed for CTS in the case of unbalanced data, however, the loss of power is always much lower than in TS. One should note that in our simulations we considered extreme values for ρ. In practice, the correlation of -0.9 is improbable. We also indicated that it is not necessarily the correlation that drives the results, since this correlation can always be made zero.

We additionally illustrated the possible danger in using the two-step approach when a SNP is cross-sectionally important, leading to either strongly inflated type I error rate or considerable loss of power.

In conclusion, the conditional two-step approach provides, in virtually all practical situations, a very good approximation of the p-value for the SNP$\times$time effect obtained from the CLMM. At the same time it protects the user better against model misspecification than the LMM and the TS. In addition, it hugely reduces demands on computing resources. The computational benefits become even more important with growing number of SNPs and in models analyzing the joint effects of multiple SNPs.

Finally, the two-step approaches can be viewed as approaches that perform inference on the longitudinal fixed effects using longitudinal summary measures, namely the random slopes. In this paper we focused on evaluating the importance of the SNPs separately, but it is clear that the CTS approach can be extended to cover mixed models with a non-linear evolution in time modeled either in a non-linear or smooth manner. The CTS approach can also be combined with other, more complex, statistical procedures in the second step evaluating the effects of the SNPs jointly.

Finally we note that our approach assumes uncorrelated measurement errors. In principle correlated errors are not covered here. However, in Jacqmin-Gadda et al. (2007) it is shown that the estimation of fixed effects is robust again violation of independence as-

sumption as long as random intercept and slope are present in the mixed model. This is also confirmed in Supplementary Figure 4. We believe that our approach therefore offers a wide range of possibly complex statistical procedures that are practically feasible with limited computational resources.

Acknowledgements

The last author acknowledges Geert Verbeke for the interesting discussions on the conditional linear mixed model.

Appendix

Linear mixed model

The ML estimator of β in model (3.1) for balanced data set can be expressed as

$$\widehat{\beta} = (\sum_{i=1}^{N} X_i^T W X_i)^{-1} (\sum_{i=1}^{N} X_i^T W y_i), \tag{3.18}$$

with $y_i = (y_{i1}, \ldots, y_{in})^T$, X_i the design matrix for the i-th subject consisting of the columns $\mathbf{1}$, $\boldsymbol{t}$, $s_i\mathbf{1}$, $s_i\boldsymbol{t}$, and W equals V^{-1}. Using Woodbury equation,

$$(A + BDB^T)^{-1} = A^{-1} - A^{-1}B(D^{-1} + B^T A^{-1} B)^{-1} B^T A^{-1},$$

W can be rewritten for model (3.2) as

$$W = \frac{1}{\sigma^2}(I_n - c\underset{\sim}{1}_n)(I_n - \gamma\, \mathbf{t}\, \mathbf{t}^T (I_n - c\underset{\sim}{1}_n)), \tag{3.19}$$

where $\mathbf{t} = (t_1, ..., t_n)^T$, is the vector of time points, $c = (\sigma^2/\sigma_0^2 + n)^{-1}$, $\gamma = (\sigma^2/\sigma_1^2 + \theta)^{-1}$ and $\theta = \sum t_j^2 - c(\sum t_j)^2$. By inserting equation (3.19) in equation (3.18) the estimate of interaction term in model (3.2), can be obtained as

$$\widehat{\beta}_3 = \frac{\text{cov}(\mathbf{s}, \mathbf{u}) - \bar{\mathbf{t}}\, \text{cov}(\mathbf{s}, \mathbf{y})}{n\ \text{var}(\mathbf{t})\ \text{var}(\mathbf{s})}.$$

By inserting W in the variance-covariance matrix of LMM, i.e. $\text{var}(\widehat{\beta}_3) = (\sum_{i=1}^{N} X_i^T W X_i)^{-1}$, the variance of the interaction estimator can be written as

$$\text{var}(\widehat{\beta}_3) = \frac{\sigma^2 + \sigma_1^2\ n\ \text{var}(\mathbf{t})}{N\ n\ \text{var}(\mathbf{t})\ \text{var}(\mathbf{s})}.$$

Two-step approach

In the second step of the two-step approach (i.e. model (3.9)) the MLE of $\widehat{\beta}_1^{**}$ can be expressed as

$$\widehat{\beta}_1^{**} = \sum \frac{(s_i - \bar{s})\hat{b}_{1i}^*}{\sum (s_i - \bar{s})^2}, \tag{3.20}$$

where $\widehat{b}_{1i}^*$ is the best linear unbiased predictor (BLUP) and can be computed by the empirical Bayesian approach as

$$\widehat{b}_i^* = DZ^T W(y_i - X\widehat{\beta}^*), \tag{3.21}$$

where $\widehat{\beta}^*$ is the ML estimator of the vector of fixed effects of reduced model (3.6). By inserting $\widehat{\beta}^*$ and W in equation (3.21) the BLUP for model (3.6) can be obtained as

$$\widehat{b}_{1i}^* = \gamma(\xi_i - \bar{\xi}), \tag{3.22}$$

where $\xi_i = \sum_j t_j y_{ij} - c(\sum_j t_j)(\sum_j y_{ij})$, $\bar{\xi} = \sum_j t_j \bar{y}_j - c(\sum_j t_j)(\sum_j \bar{y}_j)$, $\bar{y}_j = \frac{1}{N}\sum_{i=1}^{N} y_{ij}$, $c = (\sigma^2/\sigma_0^{*2} + n)^{-1}$, $\gamma = (\sigma^2/\sigma_1^{*2} + \theta)^{-1}$ and $\theta = \sum t_j^2 - c(\sum t_j)^2$. By inserting the two above equations into equation (3.20) the estimate of β_1^{**} can be derived as

$$\widehat{\beta}_1^{**} = \frac{\text{cov}(\mathbf{s}, \mathbf{u}) - n\ c\ \bar{\mathbf{t}}\ \text{cov}(\mathbf{s}, \mathbf{y})}{\text{var}(\mathbf{s})(\sigma^2/\sigma_1^{*2} + \sum_j t_j^2 - c(\sum_j t_j)^2)}.$$

Note that in the above derivations we have used the assumption that the covariance of the random intercept and slope is zero. We have argued in the text when this assumption is reasonable.

Conditional two-step approach

The transformed matrix (i.e. A) which was introduced in CLMM can be defined as

$$A =< \frac{a_1}{\parallel a_1 \parallel}, \frac{a_2}{\parallel a_2 \parallel}, \frac{a_3}{\parallel a_3 \parallel} >,$$

by using the Gram Schmidt process, where $a_1 = \mathbf{t} - < \mathbf{t} \,.\, \mathbf{1} > \frac{1}{n}$. In the second step of the conditional two-step approach (i.e. model (3.12)), the ML estimator of $\widehat{\beta}_1^{\triangle\triangle}$ can be expressed as

$$\widehat{\beta}_1^{\triangle\triangle} = \sum \frac{(s_i - \bar{s})\widehat{b}_{1i}^{\triangle}}{\sum (s_i - \bar{s})^2}, \tag{3.23}$$

where $\widehat{b}_{i1}^{\triangle}$ is the best linear unbiased predictor (BLUP) and can be computed by the empirical Bayesian approach as

$$\widehat{b}_i^{\triangle} = DZ^{*T}W^*(y_i^* - X^*\widehat{\beta}^{\triangle}), \tag{3.24}$$

where $\widehat{\beta}^{\triangle}$ is the MLE of the reduced model (3.11), X^* and Z^* are the transformed design matrices of fixed and random effects, respectively. By using the Woodbury equation, W^* can be obtained as

$$W^* = \frac{1}{\sigma^2}(\underset{\sim}{I}_n - \gamma\, Z^* Z^{*T}),$$

where $\gamma = (\sigma^2/\sigma_1^{*2} + n\ \mathrm{var}(\mathbf{t}))^{-1}$. By inserting $\widehat{\beta}^{\triangle}$ and W^* into equation (3.24) the BLUP for model (3.11), can be obtained as

$$\widehat{b}_{1i}^{\triangle} = \gamma\ (u_i - \frac{1}{N}\sum_i u_i - \bar{t}y_i + \frac{1}{N}\bar{t}\sum_i y_i),$$

with $y_i = \sum_j y_{ij}$ and $u_i = \sum_j y_{ij}t_j$. By inserting the two above equations into the equation (3.23), the ML estimator of SNP effect in model (3.12) can be derived as

$$\widehat{\beta}_1^{\triangle\triangle} = \frac{\mathrm{cov}(\mathbf{s},\mathbf{u}) - \bar{\mathbf{t}}\ \mathrm{cov}(\mathbf{s},\mathbf{y})}{\mathrm{var}(\mathbf{s})(n\ \mathrm{var}(\mathbf{t}) + \sigma^2/\sigma_1^{*2})}.$$

In the second step of the conditional two-step approach, model (3.12), variance of the $\widehat{\beta}_1^{\triangle\triangle}$ can be expressed as

$$\mathrm{var}(\widehat{\beta}_1^{\triangle\triangle}) = \frac{\mathrm{var}(\widehat{b}_{i1}^{\triangle})}{\sum (s_i - \bar{s})^2}, \tag{3.25}$$

where $\mathrm{var}(\widehat{b}_{1i}^{\triangle})$ taken from the variance-covariance matrix of LMM, i.e.

$$\mathrm{var}(\widehat{b}_i^{\triangle}) = DZ^T W^* Z^* D - DZ^T W^* X^* (X^{*T} W^* X^*)^{-1} X^* W^* Z^* D,$$

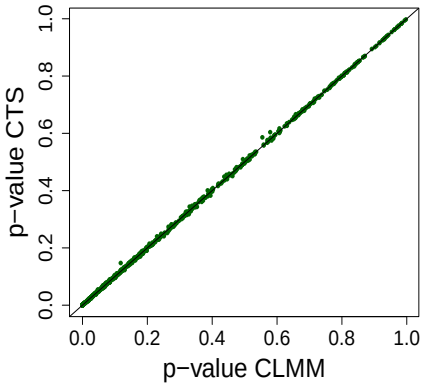
can be expressed as

$$\mathrm{var}(\widehat{b}_i^{\triangle}) = \frac{\sigma_1^{*2}}{\sigma^2}(1 - \gamma\ n\ \mathrm{var}(\mathbf{t}))\ n\ \mathrm{var}(\mathbf{t}). \tag{3.26}$$

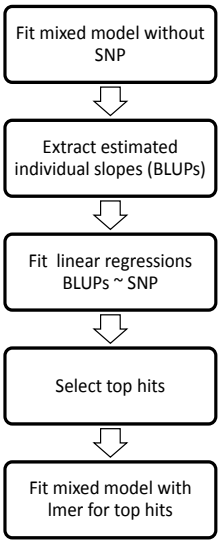
By inserting former equation into the equation (3.25), the variance of the estimate of SNP effect can be derived as

$$\text{var}(\widehat{\beta}_1^{\triangle\triangle}) = \frac{n\ \text{var}(\mathbf{t})\sigma_1^{*2}}{N\ n\ \text{var}(\mathbf{s})\ (\sigma^2/\sigma_1^{*2} + n\ \text{var}(\mathbf{t}))}.$$

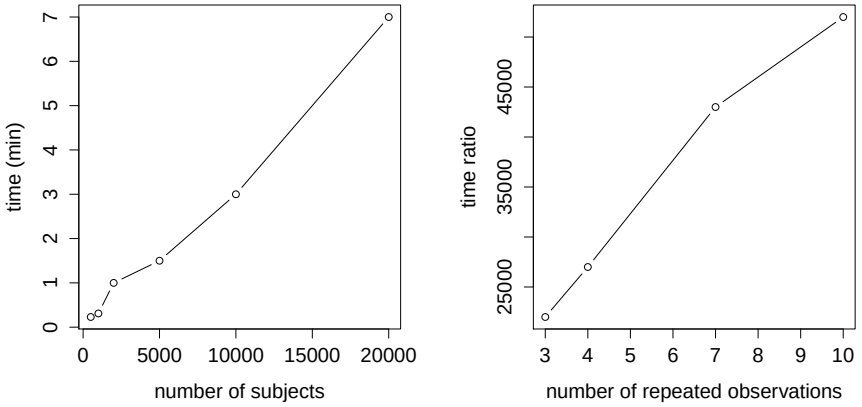
3.4 Supplementary Material



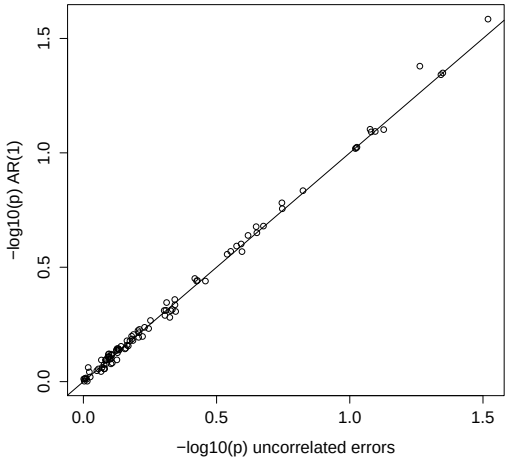
Supplementary Figure 1: MAR case, Scenario 5. Approximation of CTS comparing to CLMM.



Supplementary Figure 2: Flowchart describing practical use of CTS.



Supplementary Figure 3: Time needed to analyze 1 million of SNPs using the conditional two-step approach combined with semi-parallel regression (left panel). Computation time ratio between the function $lmer$ and the conditional two-step approach combined with semi-parallel regression depending on the number of longitudinal observations (right panel).



Supplementary Figure 4: 100 SNPs from the BMD data. On the x-axis the p-values for the SNP$\times$time interaction effect from mixed model assuming uncorrelated errors; on the y-axis the corresponding p-values from the model assuming continuous autoregressive structure for measurement error.

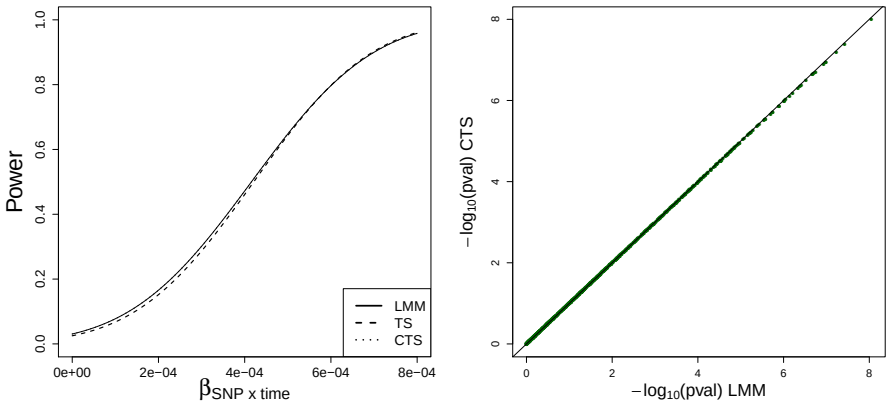


Figure: Balanced case. Performance of the approximate procedures when the time-varying covariate is included to the mixed model.

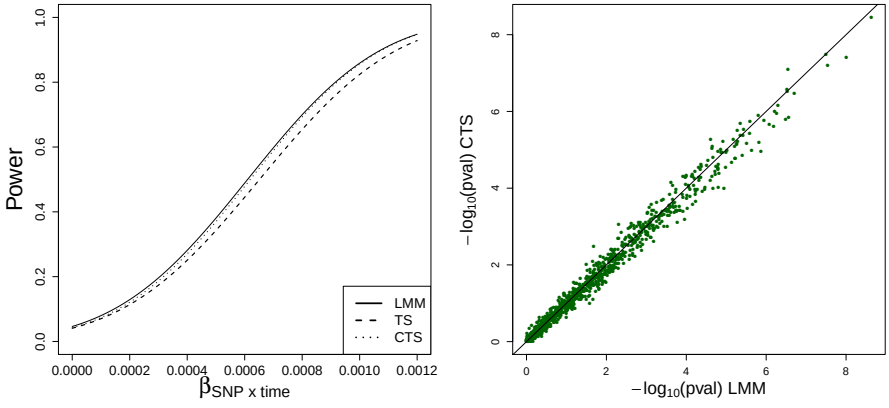


Figure: Unbalanced case. Performance of the approximate procedures when the time-varying covariate is included to the mixed model.

An example of applying the conditional two-step approach
The data are arranged in a so-called "long format" with one row per observation. The SNP data are stored in matrix S with N rows and ns columns. The size of ns depends on the available RAM. The first few rows of the phenotype data (*mydata*) look as follows:

```
   id         y Time
1   1 1.1289273    1
2   1 1.1426346    2
3   1 1.1663130    3
4   1 1.2042046    4
5   1 1.2647349    5
6   2 0.9595783    1
7   2 0.8327966    2
8   2 0.6577344    3
9   2 0.4980894    4
10  2 0.3416705    5
```

The code below, with function *cond*, transforms data for conditional linear mixed model. It is based on the SAS macro provided in Verbeke et al. in "Conditional linear mixed models" (2001). Variable "vars" is a vector with the names of the response and all the time-varying covariates that should be transformed.

```
cond = function(data, vars) {
  data = data[order(data$id), ]
  ### delete missing observations
  data1 = data[!is.na(data$y), ]
  ## do the transformations
  ids = unique(data1$id)
  transdata = NULL
  for(i in ids) {
    xi = data1[data1$id == i, vars]
    xi = as.matrix(xi)
    if(nrow(xi) > 1) {
      A = cumsum(rep(1, nrow(xi)))
      A1 = poly(A, degree = length(A)-1)
      transxi = t(A1) %*% xi
      transxi = cbind(i, transxi)
      transdata = rbind(transdata, transxi)
    }
  }
      transdata = as.data.frame(transdata)
      names(transdata) = c("id", vars)
     row.names(transdata) = 1:nrow(transdata)
     return(transdata)
}
```

The code below applies the conditional two-step approach. First, the data are transformed using function *cond*. Next, the reduced conditional linear mixed model is fitted and

the random slopes are extracted. Finally, the semi-parallel regression is performed.

```
# transform data for the conditional linear mixed model
trdata = cond(mydata, vars = c("Time", "y"))
#fit the reduced model and extract random slopes
mod2 = lmer(y ~ Time - 1 + (Time-1|id), data = trdata)
blups = ranef(mod2)$id
blups = as.numeric(blups[ , 1])
# perform the second step using semi-parallel regression
X = matrix(1, n, 1)
U1 = crossprod(X, blups)
U2 = solve(crossprod(X), U1)
ytr = blups - X %*% U2
ns = ncol(S)
U3 = crossprod(X, S)
U4 = solve(crossprod(X), U3)
Str = S - X %*% U4
Str2 = colSums(Str ^ 2)
b = as.vector(crossprod(ytr, Str) / Str2)
sig = (sum(ytr ^ 2) - b ^ 2 * Str2) / (n - 2)
err = sqrt(sig * (1 / Str2))
p = 2 * pnorm(-abs(b / err))
```

Chapter 4

GWAS on your notebook: fast semi-parallel linear and logistic regression for genome-wide association studies

Abstract

GWA studies encounter the following computational issues: a large number of tests and very large genotype files which cannot be directly loaded into the software memory. We show how to speed up the computations for cross-sectional continuous outcome using matrix operations in pure R code. Computation time from 6 hours is reduced to 10-15 minutes. Our approach can handle essentially an unlimited amount of covariates efficiently, using projections. Data files in GWAS are vast and reading them into computer memory becomes an important issue. However, much improvement can be made if the data is structured beforehand in a way allowing for easy access to blocks of SNPs. We propose several solutions based on the R packages **ff** and **ncdf**. Additionally, we adapted the semi-parallel computations for logistic regression.

Adapted version of the research article: Sikorska K., Lesaffre, E., Groenen P.J.F. and Eilers P.H.C. (2013) GWAS on your notebook: fast semi-parallel linear and logistic regression for genome-wide association studies. *BMC Bioinformatics*, 14(1), 166.

4.1 Introduction

For the benefit of readers who are not familiar with genome-wide association studies we provide a brief introduction to this area.

There are many ways to investigate the influence of genes on (human) traits. One of them, genome-wide association studies (GWAS), exploits the fact that strings of DNA contain many small variations, called SNPs which may influence the level of traits or risk of having a disease. Modern micro-array technology makes it possible to measure genotypes of a million SNPs in one go, at a reasonable price, using only one drop of blood. In large epidemiological studies, this has been done for large to very large groups of individuals, for which (many) phenotypes have been measured too. SNPs that are found to be influential may point to relevant genes. This approach has been applied on a grand scale (Pearson and Manolio, 2008). The number of results published on GWAS is rapidly increasing. The GWAS catalogue includes over 1400 papers on newly discovered important SNPs (Hindorff et al., 2010).

Typically, the number of genotyped SNPs is around half a million. However, it is possible to impute the most probable genotypes for real or hypothetical SNPs using spatial correlation on the genome. This way, the number of SNPs analyzed in a GWAS can grow to 2.5 or even 30 million.

The statistical model used in GWAS is rather basic: univariate linear or logistic regression of phenotype on genotypes, for each SNP in turn, correcting for covariates like age, height and gender. Large sample sizes are required to detect very small effects at the very strict "GWA-significance level", namely 5×10^{-8}, the common 0.05 divided by one million (inspired by Bonferroni correction for that many tests). The goal is to find SNPs for which the p-value will survive this conservative multiple testing correction.

Dedicated software is available to support those analyses. Popular examples are: GenABEL (Aulchenko et al., 2007), PLINK (Purcell et al., 2007), Mach2qtl (Li et al., 2010, 2009) and ProbABEL (Aulchenko et al., 2010). Computation times are long. An example from the literature is a GWAS with a continuous trait for 6000 individuals and 2.5 mln SNPs, which on "a regular computer" takes around 6 hours (Estrada et al., 2009). This time will dramatically increase with larger sample size and/or more SNPs to test. Additionally, logistic regression is more computationally demanding than linear regression. Based on the available published materials, it is actually quite difficult to assess computation times. Usually information about available memory, number of used processors/cores, and the size of the model (the number of covariates) are not provided.

GWAS may be computationally demanding, but the problem is "embarrassingly parallel", meaning that it can be distributed over as many processors as desired, by simply splitting the work into groups of SNPs. This brute force approach with computing clusters is now being applied broadly, with GRIMP (Estrada et al., 2009) as an example in our institution.

We show that huge speed gains can be achieved by simple rearrangements of linear model computations, exploiting fast matrix operations. We call this the "semi-parallel" approach, to set it apart from parallel computation on multiple processors. A similar idea can be found in Shabalin (2012), in the framework of expression quantitative trait loci (eQTL) analysis. That paper focuses on computing R^2 statistics, to get a first insight into a

data. We are more ambitious: we want to reproduce very closely the results of "traditional" GWAS software. Present-day GWAS practice is focused on very low p-values, regardless of the amount of variance that the SNPs actually explain. Thus, we apply large matrix operations to compute estimates, standard errors and p-values for GWAS with a continuous outcome.

There is a second challenge: reading the data quickly enough from a disk into computer memory. A key issue is to rearrange them in such a way that arbitrary blocks of SNPs (containing all individuals) can be accessed very quickly. We show how to pre-process data for this goal.

The bottom line is that a GWAS for one million SNPs and 10k individuals can be done on an average notebook computer within 15 minutes. This is the time needed for pure computations. Accounting for the time needed to load the data, the whole time of the analysis increases to 25 minutes.

Semi-parallelization of GWAS with a binary outcome is more difficult. Parameters in logistic regression are estimated via maximum likelihood, which unlike the least squares approach is an iterative procedure. However, we were able to find an approximate way to provide odds ratio for the SNP effect using semi-parallel computations.

The paper is written in a tutorial-like manner. We gradually extend the complexity of the problem, showing step by step how to speed up computations using simple tricks in R. Also the goal is not to present a package (there is none) but to introduce a new way of thinking about large-scale GWAS computations and to present and provide code that anyone can easily integrate into existing systems.

4.2 Implementation

Data, real and simulated

A GWAS is based on very large numbers of SNPs, for many thousands of individuals, leading to very large data files. Observed genotypes generally are coded as the number of reference alleles, 0, 1 or 2. Very efficient storage is possible, using only 2 bits per SNP (per individual). The program PLINK uses this approach to store genotypes in its BED file format. The package SNPstats mimics it for storing SNPs in computer memory. This is quite attractive: 100k SNPs (we use k as shorthand for thousand) for 10k individuals can be stored in a quarter of a Gigabyte.

In large data sets one may expect some values of the traits and/or genotypes to be missing. Typically genotypes with a call rate (percentage of measured genotypes in the sample) below 95% will be removed from the analysis.

The recent GWAS practice is to use genotype imputation. The commonly used MACH program does two things: it imputes missing SNPs within genotyped markers and predicts untyped markers. The result of imputation is the expected dose, a non-integer number between 0 and 2. In this case more room is needed to store the values. Actual files with imputation results are then much larger. Those that MACH produces are ASCII-code files, using six positions per number (with three decimals). In principle a more compact representation is possible. There is no need to be precise and by multiplying the dose by 100 we

can store an integer between 0 and 200 in one (unsigned) byte. This is four times larger than for raw genotypes.

In our experience, SNP data are stored in such a way that all SNPs for one individual form one record. We call this structure "row per person". It is useful for random reading of (blocks of) individuals, but selection of certain (blocks of) SNPs is time consuming. Essentially one has to read the complete records for all persons and keep only the required selection of SNPs. This has to be repeated for every block of SNPs one considers.

It is much more attractive to have each record represent one SNP, as measured for all individuals ("row per SNP"). To achieve this, given a "row per person" organization, is an important part of the enterprise. It would not be an issue if all data would fit into fast random-access memory, but this is usually not the case, as we are talking of 10 to 100 GB.

There is no need for real data when discussing computation times. Instead we simulate genotypes as random numbers from a uniform distribution between 0 and 2. Phenotypes and covariates are simulated as independent variables coming from a standard normal distribution. In our simulations we set the sample size to a typical GWAS scenario, namely 10K individuals. The number of simulated SNPs is 1000, which is determined by the available RAM.

Semi-parallel computations

In this section we present semi-parallel computation, using the R programming language as the vehicle for implementation (R version 2.15). We report computation speeds, as achieved on a single PC running Windows XP on an Intel E8400 (3.00 GHz) with 3.2 GB of RAM. We report user times provided by the R function `proc.time`. User time is defined as the CPU time charged for the execution of user instructions of the calling process.

A simple benchmark for comparing to other computers is the time needed for the singular value decomposition (SVD) of a 1000×1000 random matrix. For our computer it is 5 seconds in R software.

Our goal is to report computation times in a standardized way, such that they can be easily recalculated for different numbers of individuals and/or SNPs. Computation time for GWAS is linear in sample size and in the number of investigated SNPs. We express speed in "sips" standing for "snp-individual per second". It is obtained by dividing the product of the numbers of individuals and SNPs by the time needed for a computation. Conversely, if one divides the product of the number of individuals and the number of SNPs by speed, one obtains the number of seconds needed for a job. One should keep in mind that due to the imprecision of `proc.time` and its variability from run to run, the calculated times/speeds are only approximate. They are provided to assess the order of magnitude of the times gained in computations. Because of the size of the numbers, we will exclusively use Msips, meaning one million sips.

Regression without additional covariates

Let the (continuous) phenotype be given as a vector y of length n and the states of m SNPs as the $n \times m$ matrix S. A single column of S will be denoted as s. Unless stated otherwise,

we use the same symbols for the R variables. To detect potential genetic effects on y, the linear model

$$y = \alpha + \beta s + \epsilon \tag{4.1}$$

is fitted for each SNP and the size of $\hat{\beta}$ is evaluated. Generally the estimated effects are disappointingly low. A culture has grown in which one searches for low (Bonferroni corrected) p-values, using large to very large sample sizes. To compute p-values we need standard errors, but we will not consider them until the model with covariates has been discussed. A straightforward way to fit the model (4.1) is to use the function `lm` repeatedly.

```
t0 = proc.time()[1]
beta = rep(0, m)
for(i in 1 : m) {
   mod = lm(y ~ S[ , i])
   beta[i] = mod$coeff[2]
   }
t1 = proc.time()[1]-t0
cat("Speed", 1e-6 * n * m/t1, "Msips\n")
```

The reported speed is 0.8 Msips, meaning that for this sample size we can test 80 SNPs per second. For 2.5 M SNPs we would need almost 9 hours. In the code above we included the statements used to compute processing times and speed. They will not be shown in the upcoming examples. A faster alternative to `lm` is `lsfit`, recording a speed of 5.3 Msips.

```
beta = rep(0, m)
for(i in 1 : m){
  mod = lsfit(S[ , i], y)
  beta[i] = mod$coeff[2]
}
```

For this simple regression problem, we know how to compute the slope explicitly:

$$\hat{\beta} = \frac{\sum_{i=1}^{n}(s_i - \bar{s})(y_i - \bar{y})}{\sum_{i=1}^{n}(s_i - \bar{s})^2}. \tag{4.2}$$

This is implemented in the following code, which increases the speed to 26 Msips.

```
beta = rep(0, m)
yc = y - mean(y)
for(i in 1 : m){
  sc = S[ , i] - mean(S[ , i])
  beta[i] = sum(sc * yc)/(sum(sc ^ 2))
  }
```

So far, we considered cases where the analysis is implemented in a loop, for one SNP at a time. However, loops are inefficient and it is better to vectorize the computations. That leads us to our first *semi-parallel* algorithm. In the previous code fragment we took each column of the SNP matrix, to center it and compute its inner product with centered y, $\tilde{y}$. If we center all columns at once using the function `scale` we obtain the whole vector $\hat{\beta}$ without using loops. However, when running the code below

```
yc = y - mean(y)
Sc = scale(S, scale = T)
S2 = colSums(Sc ^ 2)
b = crossprod(yc, Sc) / S2
```

we get an unpleasant surprise: the speed drops to 19 Msips. It turns out that `scale` is a very slow function. We were able to avoid it when we did the calculations ourselves and achieved a speed of 45 Msips.

```
yc = y - mean(y)
s1 = colSums(S)
e = rep(1, n)
Sc = (S - outer(e, s1 / n))
b = crossprod(yc, Sc) / colSums(Sc ^ 2)
```

Centering columns of the SNP matrix is actually not necessary. We can rewrite the numerator of the equation (4.2) as

$$\sum_i^n \tilde{y}_i(s_i - \bar{s}) = \sum_i^n \tilde{y}_i s_i - \bar{s}\sum_i^n \tilde{y}_i = \sum_i^n \tilde{y}_i s_i.$$

Similarly, we can show that the denominator of (4.2) can be rewritten as

$$\sum_i^n (s_i - \bar{s})^2 = \sum_i^n s_i^2 - n(\bar{s})^2.$$

This leads to the following code, running at 90 Msips.

```
yc = y - mean(y)
s1 = colSums(S)
s2 = colSums(S ^ 2)
b = crossprod(yc, S) / (s2 - (s1 ^ 2) / n)
```

This means that to analyze a GWAS with 2.5 mln of SNPs and 10k individuals around 5 minutes are needed. However, this is for an unrealistic scenario, without covariates. Also, we have not calculated the p-values yet. We will now discuss the needed extensions.

Regression with covariates

To handle covariates in a matrix X, we extend the model (4.1) to

$$y = \beta s + X\gamma + \epsilon, \tag{4.3}$$

where it has been assumed that X contains a column of ones, to cater for an intercept. A straightforward application of this model uses `lm` in a loop, as shown below.

```
for(i in 1:m){
  mod = lm(y ~ S[,i] + X - 1)
  b[i] = summary(mod)$coeff[1]
}
```

Of course the speed will now depend on the number of covariates. This relation is shown in Figure 4.1.

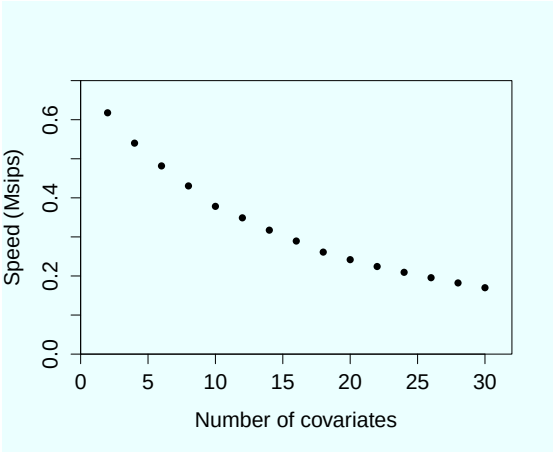


Figure 4.1: The plot shows the relationship between the speed of the computations using `lm` function in R and the number of the covariates in the linear regression model.

We can also repeatedly apply function `lsfit` in the following manner.

```
b = rep(0, m)
for(i in 1 : m){
  mod = lsfit(cbind(S[ , i], X), y, intercept = F)
  b[i] = mod$coeff[1]
}
```

For 10 covariates, speed is equal to 1.17 Msips. It is again faster than `lm`, but the whole GWA scan for 2.5 mln SNPs and 10K individuals would still take around 6 hours (18 hours for `lm`). Shabalin (2012) has briefly discussed how to deal with one covariate in an efficient way. The main idea is to orthogonalize the response and the predictor of interest (here the SNP) with respect to that covariate. We derived it for the general case with k covariates (see Appendix). The transformed variables are given by the equations:

$$s^* = s - X(X^T X)^{-1} X^T s, \tag{4.4}$$

$$y^* = y - X(X^T X)^{-1} X^T y. \tag{4.5}$$

Assuming that the intercept was included in the matrix of covariates, the model is now simplified to

$$y^* = \beta s^* + \epsilon. \tag{4.6}$$

It is important to calculate y^* and s^* efficiently. If we multiply the matrices in order as they appear in (4.4) and (4.5), R will encounter memory problems when working with $n \times n$ matrix. A code fragment for well-organized calculations is shown below.

```
X = cbind(1, X0)
U1 = crossprod(X, y)
U2 = solve(crossprod(X), U1)
ytr = y - X %*% U2
U3 = crossprod(X, S)
U4 = solve(crossprod(X), U3)
Str = S - X %*% U4
b = as.vector(crossprod(ytr, Str) / colSums(Str ^ 2))
```

Speeds are 45, 25, 13 Msips for 2, 10 and 30 covariates respectively, about 70 times faster than using `lm`.

Standard errors and p-values

The variance for the estimated $\hat{\beta}$ in model (4.6) is given by

$$\widehat{\text{var}}(\hat{\beta}) = \hat{\sigma}^2 (s^{*T} s^*)^{-1}. \tag{4.7}$$

The error variance is estimated by

$$\hat{\sigma}^2 = \frac{\text{RSS}}{n-k-2}, \tag{4.8}$$

where the residual sum of squares (RSS) is calculated as

$$\text{RSS} = (y^* - s^* \hat{\beta})^T (y^* - s^* \hat{\beta}) = \sum_i^n y_i^{*2} - \hat{\beta}^2 \sum_i^n s_i^{*2}. \tag{4.9}$$

Note that in the degrees of freedom we have accounted for the removed covariates, although this usually will be of minor influence. The standard errors of $\hat{\beta}$ and logarithm of the p-values can be calculated with the code below.

```
Str2 = colSums(Str ^ 2)
sig = (sum(ytr ^ 2) - b ^ 2 * Str2) / (n - k - 2)
err = sqrt(sig * (1 / Str2))
p = 2 * pnorm(-abs(b / err))
logp = -log10(p)
```

Table 4.1: Speed in Msips for linear model (estimates, standard errors and p-values) with k covariates for the functions `lm`, `lsfit` and semi-parallel (SP).

k	lm	lsfit	SP
0	0.70	3.0	43.0
2	0.60	2.4	43.0
10	0.40	1.0	25.0
30	0.16	0.32	12.0

The calculation of the p-values assumes, given the large sample size, that the test statistic has a normal distribution. We used the lower tail of the normal distribution to calculate

the p-values. It is not advisable to use the textbook definition `2 * (1-pnorm(b / err))`, because it suffers from severe rounding errors.

We make a final comparison of speed between `lm`, `lsfit` and our fast computations. Standard errors and p-values are not included in the `lsfit` function, but are easily obtained those using `ls.print` procedure. The results, for different numbers of covariates, are provided in the Table 4.1. We see that the standard function `lm` is the slowest, but the computational benefits of `lsfit` decrease for the cases with many covariates. Using semi-parallel computations, we can do a GWAS 61 times faster than with `lm` for no covariates and 75 times faster for a model with 30 covariates. A GWA scan for 10K individuals, 2.5M SNPs and 10 covariates can be now done within 20 minutes.

We tested our codes on a another PC with Intel Xeon(R) X5550, 2.67 GHz, 24 GB of RAM and the 64 bit version of R. This machine was around 1.4 times faster than our PC. However, the ratios of the speeds remained similar. Semi-parallel approaches is 60-80 times faster than looping function `lm`.

Missing genotypes

The semi-parallel algorithm does not allow missing values. A single `NA` in either a phenotype vector or a SNP matrix will result in `NA` in the whole vector of estimates.

Incomplete phenotypes are easy to handle. We can exclude those individuals from the whole analysis. Missing genotypes are more problematic. In general missing data can be handled using weighted least squares estimation, taking as weights 0 and 1 for missing and available observation. However the weights will vary for different SNPs and semi-parallel approach for the model with covariates cannot be applied anymore.

We propose a very simple solution for the analysis of a GWAS with incomplete SNPs by imputing the missing SNP values with the sample mean of the observed genotypes. Our simulations show that for large sample size (thousands of individuals) and even 5% missing genotypes no substantial precision is lost (Figure 4.2).

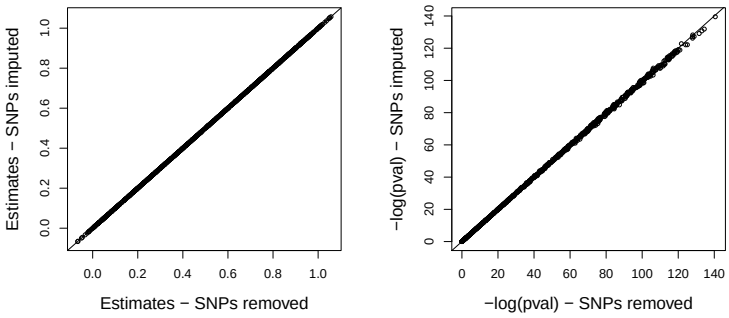


Figure 4.2: The effect of imputation of missing SNPs using sample mean on estimates and p-values

Logistic regression

When there is an interest in association between a binary outcome and SNPs, logistic regression is needed. The model without additional covariates is given by

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 s, \tag{4.10}$$

with p representing probability of "success".

No closed-form expression exists for the coefficient values. Instead, the (logarithm of the) likelihood function is maximized using iterative procedures like Newton-Raphson or Fisher scoring. The maximization begins with a tentative solution which is iteratively improved until convergence. In R the straightforward way to fit a logistic model is to call a function fitting generalized linear model specifying proper outcome distribution (binomial) and link function (logit). It can be easily done using the code

```
beta = rep(0, m)
for (i in 1:m) {
  mod = glm(y ~ S[,i], family = binomial ("logit"))
  beta[i] = mod$coeff[2]
}.
```

The speed is 0.2 Msips which is four times slower than fitting a regression model to a continuous outcome.

A relation exists between maximum likelihood estimation using Fisher scoring and weighted least squares estimation (Agresti, 2002). Maximum likelihood equations for the $(t+1)$-th iteration can be written as

$$(X^T W^{(t)} X)\beta^{(t+1)} = X^T W^{(t)} z^{(t)}, \tag{4.11}$$

where z is a "working variable" given by

$$z_i^{(t)} = \log\left(\frac{p_i^{(t)}}{1-p_i^{(t)}}\right) + \frac{y_i - p_i^{(t)}}{w_i^{(t)}} \tag{4.12}$$

and where $W^{(t)}$ is diagonal matrix with elements $p_i^{(t)}(1-p_i^{(t)})$. Every update of β involves solving a weighted least squares problem with updated weight matrix. This process is called iteratively reweighted least squares. The covariance matrix is given by

$$\widehat{\text{cov}}(\hat{\beta}^{(t+1)}) = (X^T W^{(t)} X)^{-1}. \tag{4.13}$$

In case of a model without additional covariates the estimated SNP effect and the standard error are given by

$$\widehat{\beta_1} = \frac{\sum_i w_i (z_i - z_w)(s_i - s_w)}{\sum_i w_i (s_i - s_w)^2}, \tag{4.14}$$

$$\widehat{\text{var}}(\beta_1) = \frac{1}{\sum_i w_i (s_i - s_w)^2}, \tag{4.15}$$

where z_w and s_w are weighted means defined as $\sum_i w_i z_i / \sum_i w_i$ and $\sum_i w_i s_i / \sum_i w_i$, respectively.

It is not possible to semi-parallelize logistic regression computations to provide an exact solution, because in principle the weights are different for each SNP. However, effects found in GWAS are usually of modest size, with a median odds ratio of 1.33 and only a few odds ratios exceeding 3.00 (Feero et al., 2010). This means that probabilities predicted by a model without a SNP will not change much once SNP is included to the model. We can do semi-parallel computations approximately using weights from the model without SNP ($\tilde{w}$) as starting values and updating the solution for β_1 by one iteration. Note that in case of no other covariates we have to fit the model with only intercept. The predicted probabilities are the same for every individual and so are the weights. In that special case the weighted mean is equal to the arithmetic mean and (4.14) reduces to (4.2). The computations can be easily done in R using the code

```
#### fit model without SNP and set weights
mod0 = glm(y ~ 1, family = binomial ("logit"))
p = mod0$fitted
w = p * (1 - p)
### Do the computations
z = log(p / (1 - p)) + (y - p) / (p * (1 - p))
zc = z - mean(z)
s1 = colSums(S)
s2 = colSums(S ^ 2)
den1 = s2 - s1 ^ 2 / n
b = crossprod(zc, S) / den1
err = sqrt(1 / (w[1] * den1))
pval = 2 * pnorm(-abs(b / err)).
```

The speed is 55 Msips which is 275 faster than using `glm`.

Obviously, the quality of approximation of the weights from the model without the SNP depends on the magnitude of β_1. We conducted a small simulation experiment exploring an effect of a true odds ratio on the accuracy of estimation in semi-parallel approach. We simulated 1000 logistic regression models in which true OR was a random number between 1 and 5. We calculated the relative difference of the odds ratios estimated by `glm` function and semi-parallel approach. The relative difference is increasing monotonically and non-linearly with the correctly estimated OR (Figure 4.3). The semi-parallel approach underestimates the OR by 0.1% for OR = 1.33, by 6% for OR = 3 and by 17% for OR = 5. This result was independent from the sample size. Additionally, the p-values in semi-parallel approach were slightly too significant, but the difference was observed only for the $-\log_{10}(p)$ above 25 (for the sample size 2000, Figure 4.4). We do not find those observations worrisome in a typical GWAS scenario. However, we leave it up to the user to additionally fit the `glm` model to a selection of the most promising SNPs.

Dealing with covariates in semi-parallel logistic regression follows the same reasoning as in linear regression, but taking the weight matrix into account. The equations for trans-

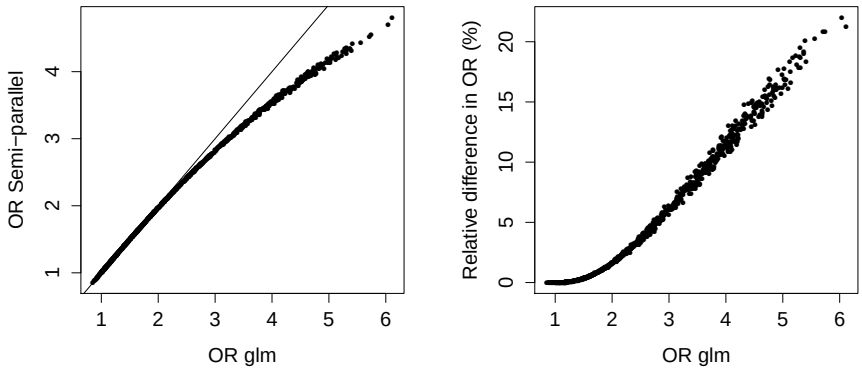


Figure 4.3: Odds ratios from the standard approach (`glm`) versus semi-parallel approach.

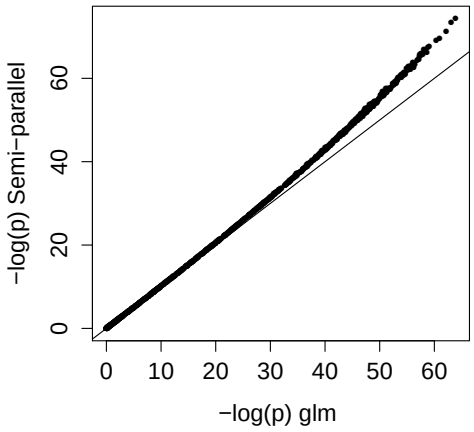


Figure 4.4: P-values from the standard approach (`glm`) versus semi-parallel approach.

formed SNP (s^*) and z^* are

$$s^* = s - X(X^T W X)^{-1} X^T W s, \tag{4.16}$$

$$z^* = z - X(X^T W X)^{-1} X^T W z, \tag{4.17}$$

where again X is a matrix of covariates including an intercept. The weight matrix W is replaced with $\tilde{W}$ coming from the model without SNP. After the transformation the solution for SNP effect and the standard error are given then by

$$\widehat{\beta_1} = \frac{\sum_i w_i z_i^* s_i^*}{\sum_i w_i s_i^{*2}}, \tag{4.18}$$

and

$$\widehat{\text{var}}(\beta_1) = \frac{1}{\sum_i w_i s_i^{*2}}. \tag{4.19}$$

Noting that in this case weights are different for every individual, we can compute the solution by running the following R code

```
mod0 = glm( y ~ X, family = binomial("logit"))
p = mod0$fitted
w = p * (1 - p)
z = log(p / (1 - p)) + (y - p) / (p * (1 - p))
xtw = t(X * w)
U1 = xtw %*% z
U2 = solve(xtw %*% X, U1)
ztr = z  - X %*% U2
U3 = xtw %*% S
U4 = solve(xtw %*% X, U3)
Str = S - X %*% U4
Str2 = colSums(w * Str^2)
b = crossprod(ztr * w, Str)/Str2
err = sqrt(1/ Str2)
pval = 2 * pnorm(-abs(b / err)).
```

Comparisons of speeds between semi-parallel approach and `glm` for different number of covariates are presented in Table 4.2. The speed gains are between 80 times for the model with 30 covariates and 170 times for the model with 10 covariates making the efficiency even larger than in linear regression.

Table 4.2: Speed in Msips for logistic model (estimates, standard errors and p-values) with k covariates for the functions `glm` and semi-parallel (SP).

k	glm	SP
1	0.2	20.0
10	0.1	17.0
30	0.1	8.0

4.3 Organization of the SNP data

Our semi-parallel algorithms substantially reduce computation times. However before we can apply our algorithm we need to load the data into computer memory. Of course this always is an issue, but not really critical when computations are slow.

We assume that we have limited memory available, say 2 to 4 GB. With 64 bit operating systems, 64 bit R and expensive hardware, it is possible to build a system that can have all data in memory. We do not expect the reader to be that lucky. Instead we assume that we will read in blocks of SNPs of reasonable size.

Data loading entails not only CPU but also I/O times. That is why in this section we only focus on the elapsed time provided by `proc.time`. This is the clock time measured from the start of the operation until its completion.

We propose different solutions depending on the type of the genotypes we are dealing with (observed or imputed). We show how to efficiently deal with PLINK data formats. For imputed dosage (MACH) files, we discuss what the difficulties are when loading in the structure necessary to apply fast computation algorithm. We describe two R packages: **ff** (Adler et al., 2012), **ncdf** (Pierce, 2011) which we found most useful to tackle this problem.

Observed genotypes in PLINK format

As an example, we utilized a PLINK BED file that we encountered at our institution. This file stores around 42000 SNPs on chromosome 1 measured for about 6000 persons. Some values were coded as missing. We can easily read a PLINK BED file into R using the function `read.plink` implemented in the package **SNPstats** (Clayton, 2012).

```
library(SnpStats)
P = read.plink("Chrom-01.bed")
U = P$genotypes@.Data
```

This will store the genotypes in a SnpMatrix raw format. It is a very efficient storage scheme, using only 2 bits for each element of the SNP matrix. It takes around 10 seconds to load the data. Of course, this can be done only if the matrix fits in memory, but that is no problem here. Another useful feature of the SnpMatrix object is that the indexing operator returns a matrix. Having a SnpMatrix object, we can extract blocks of SNPs to a floating point matrix. The maximum allowed size of the block depends on the available memory and the operating system.

```
#### Read blocks of SNPs
for (k in 0:nb) {
j1 = 1 + k * bs
j2 = min(j1 + bs - 1, m)
cat("Block", k, "of", nb, '\n')
if (j2 > j1) {
# Read SNPS
  S = as.numeric(U[, j1 : j2]) ## returns a vector
  ns = j2 - j1 + 1
```

```
  dim(S) = c(n, ns)
}
```

Another problem that we have to deal with are missing genotypes, but in the previous section we proposed a simple strategy to overcome this problem. In the code given below we assume that the missing values are coded as 3 and that the threshold for the call rate is 5%.

```
# Remove bad SNPs (missing SNPs are coded as 3)
fmis = colSums(S == 3) / n
sel = fmis < 0.05
S = S[, sel]
W = S < 3
# Impute missing SNPs
imp = colSums(W * S) / colSums(W)
Imp = outer(rep(1, n), imp)
S = W * S + (1 - W) * Imp
```

Once "bad" SNPs have been removed we can apply a fast computation algorithm. We analyzed a model with correction for 25 covariates. The association scan for our example data file was finished within one minute.

Imputed genotypes in MACH format

MACH files are larger than PLINK files and may include hundreds of thousands of SNPs written as "row per person" in text files. On a computer without large amount of RAM we will not be able to read into R the whole data file. We have to work with blocks of SNPs. The "row per person" structure is very inefficient if we want to read only a group of SNPs (say 1000) for all individuals. Having a transpose of it, so the "row per SNP" would make it possible for function `scan` to create a matrix with a block of SNPs for all persons. But even then, reading 1000 SNPs for 10000 individuals takes around 13 seconds. For a genome with 2.5 mln SNPs we would need around 9 hours just to bring the data into R.

There are other, faster ways to deal with large data files in R. One possibility is to work with binary files. Saving and reading binary files is easily done using `writeBin` and `readBin`. However, those files work on vectors. This is not an optimal solution for us. Saving all the genotypes for individuals sequentially will not allow us for an easy access to the blocks of SNPs later on.

There are several packages available which deal with array oriented binary files. We will discuss here **ncdf** and **ff** which we found the most useful. The Network Common Data Form (netCDF) are commonly used in meteorology and oceanography. Recently the R package **ncdf** was released to support this data format (Pierce, 2011). First, the MACH data files have to be saved into a **ncdf** object. Our experiments showed that it is most efficient to work with blocks of SNPs and individuals. We will denote `bsx` and `bsy` as block size for individuals and SNPs respectively. Number of blocks will be denoted as `nbx` and `nby`. We need to define dimensions and variables of the **ncdf** object.

```
# Define dimensions
dimx = dim.def.ncdf("x", "units", 1:nx)
dimy = dim.def.ncdf("y", "units", 1:ny)
# Define variables
varz = var.def.ncdf("z", "nix", dim = list(dimx, dimy),
                    missval = 999, prec ="short" )
```

For the specified variables, a netCDF file is created using

```
# Create the netCDF file
netf = create.ncdf(fname, vars = list(varz)).
```

If Z is a bsx$\times$ bsy block read by the function scan, the data can be easily stored into **ncdf** file.

```
# Read blocks and store them
for(i in 1:nbx) {
  k = 1 + (i - 1) * bsx
  put.var.ncdf(netf, varz, vals = Z,
               start = c(k, 1), count = c(bsx, ny))
  cat('Block', i, '\n')
}
# Close the file
close(netf)
```

Saving 100000 SNPs for 6000 persons would take around 45 minutes on our computer. To estimate the complete time, we need to add the time needed for scanning the MACH file (about 15 minutes). To read back the file created above (with the same block sizes) we have to use the following code

```
netf = open.ncdf(fname)
for (i in 1:nby) {
  k = (i - 1) * bsy + 1
  Z = get.var.ncdf(netf, varz, start = c(1, k) , count = c(nx, bsy))
}
close(netf).
```

Reading goes very fast and is done within 30 seconds.

The package **ff** was created to support memory efficient storage of the large data files. Keeping notation from the **ncdf** example, an **ff** object creation and data storage are done using

```
FF = ff(vmode = "short", dim = c(nx,ny), filename = fname )
for(i in 1 : nbx) {
  k = 1 + (i - 1) * bsx
  FF[k:(k + bsx - 1), ] = Z
  cat('Block', i, '\n')
}
close(FF).
```

After the object is created, the R workspace should be saved. The data saving is faster in **ff** than in **ncdf**. It is linear with the number of individuals's block for the fixed number of SNPs. We recorded less than a minute necessary to save 100000 SNPs for 6000 persons. To read back the blocks we need to load the saved R workspace. This workspace keeps the pointer to the **ff** file. After that, data reading is very straightforward.

```
for (i in 1:nby) {
  k = (i - 1) * bsy + 1
  Z = FF[ , k : (k + bsy - 1)]
  }
```

The elapsed times for reading are as similar to those of **ncdf**.

4.4 Results and Discussion

Computations for GWAS were made easy. We have shown that they can be rearranged as large matrix operations performing 60-80 times faster for linear regression and up to 300 times faster for logistic regression. The algorithms can be written in pure R and they do not exceed 20 lines of code.

Fast computations demand fast access to the data and this is actually a harder problem. Not all SNP data fit in memory at the same time. They have to be read in as blocks containing all individuals and selections of SNPs. In practice data are not organized in this way, but as records that contain all SNPs for each individual. We have shown two ways to rearrange data, in a preliminary step, to make fast access possible. Our first solution uses the standardized netCDF file format. It has the advantage that the files can be exchanged easily between computers, operating systems and programming languages. Our second solution uses memory mapped files, as implemented in the package **ff**. It is the fastest solution and it is easy to use, but it is less portable than netCDF.

We believe that we have presented here an attractive solution to computations for relatively large GWAS, on modest hardware, using pure R code. Our algorithms are still "embarrassingly parallel": it is trivial to divide the task over multiple machines, each working on a different block of SNPs. However, using the package SNOW to exploit multiple processors in one PC, we discovered that it takes so much time to load the data into separate processes that it was not worth the effort.

Using the many processors on modern graphic cards looks like an attractive road to explore. We feel that we are still in a transition phase in which easily accessible libraries for R are not yet available. At the moment of writing this manuscript, most available packages are tied to Nvidia GPUs and needed special installation procedures. We have not yet explored this approach.

A more complicated case of weighting is encountered when one corrects for correlation between individuals. Because the relationship matrix has as many rows and columns as the number of individuals, this poses a real challenge. Several solutions have been proposed, see e.g Lippert et al. (2011). More research is needed to determine whether they can be combined with our semi-parallel approach.

GWAS for static phenotypes is only one important issue. Much more challenging are longitudinal data, in which multiple measurements per individual are available. In general the number of measurements varies between persons, as well as the times of observation. One has to use linear mixed models, which entail heavy computation loads. A typical mixed model with 10K observations takes about 1 second, implying a speed of 0.01 Msips, more than 100 times slower than a linear model. The need for fast computations in case of longitudinal data and few approximate procedures have been described in Sikorska et al. (2013b). We are working on algorithms involving large matrix operations for massive fitting of linear mixed models. We have had some successes, but a lot has still to be done. We will report on this subject in due time.

APPENDIX

The minimum least squares fit for the model

$$y = \beta s + X\gamma + \epsilon, \tag{4.20}$$

is obtained by solving the linear system of equations

$$\begin{pmatrix} s's & s'X \\ X's & X'X \end{pmatrix} \begin{pmatrix} \hat{\beta} \\ \hat{\gamma} \end{pmatrix} \begin{pmatrix} s'y \\ X'y \end{pmatrix} \tag{4.21}$$

for $\hat{\beta}$ and $\hat{\gamma}$. This system is equivalent to the two equations

$$s^T s\hat{\beta} + s^T X\hat{\gamma} = s^T y, \tag{4.22}$$

$$X^T s\hat{\beta} + X^T X\hat{\gamma} = X^T y. \tag{4.23}$$

From equation (4.22) it follows that

$$\hat{\gamma} = (X^T X)^{-1}(X^T y - X^T s\hat{\beta}). \tag{4.24}$$

If we insert that into the equation (4.23) equation we get

$$(s^T s - s^T X(X^T X)^{-1} X^T s)\hat{\beta} = s^T y - s^T X(X^T X)^{-1} X^T y. \tag{4.25}$$

We introduce new SNP variable s^* and new outcome y^* defined as

$$s^* = s - X(X^T X)^{-1} X^T s, \tag{4.26}$$

$$y^* = y - X(X^T X)^{-1} X^T y. \tag{4.27}$$

Note that the transformed variables have clear a interpretation. They are part of s and y that is orthogonal to the space spanned by the columns of X, since $X(X^T X)^{-1} X^T s$ and $X(X^T X)^{-1} X^T y$ are the projections of s and y on X.

Now equation (4.25) has the form

$$s^T s^* \hat{\beta} = s^T y^*. \tag{4.28}$$

We can easily calculate the following

$$\begin{aligned} s^{*T} s^* = [s^T - s^T X(X^T X)^{-1} X^T][s - X(X^T X)^{-1} X^T s] = \\ = s^T s - s^T X(X^T X)^{-1} X^T s = s^T s^*. \end{aligned} \tag{4.29}$$

Similarly we can get that $s^{*T} y^* = s^T y^*$. We can now write the equation (4.28) as

$$s^{*T} s^* \hat{\beta} = s^{*T} y^*. \tag{4.30}$$

Therefore, $\hat{\beta}$ is a solution of a new regression model

$$y^* = \beta s^* + \epsilon, \tag{4.31}$$

with simpler solution (as the new model has no intercept)

$$\hat{\beta} = \frac{\sum_i^N s_i^* y_i^*}{\sum_i^N s_i^{*2}}. \tag{4.32}$$

Chapter 5

More GWAS on your notebook: fast mixed models for longitudinal phenotypes using QuickMix

Abstract

Although genome-wide association studies on cross-sectional outcomes are extremely popular, longitudinal data are hardly being explored. Computation time is a limiting factor since a typical GWA scan for 1 million of SNPs takes around 2 weeks on a single computer. We present an algorithm, *QuickMix*, to reduce that time substantially. The key idea behind our method is to write a mixed model as penalized least squares and assume that the penalty matrix is known. Furthermore, we explore the structure of the equations to avoid operations on large matrices. *QuickMix* reduces the computations from 2 weeks to half an hour on a notebook computer. Its performance is excellent in the range of p-values encountered in GWAS, providing almost exact estimates of both cross-sectional and longitudinal SNP effects. *QuickMix* perfectly combines speeding up with accuracy. It facilitates GWA analysis of longitudinal data on a single everyday computer within reasonable time frame.

Adapted version of the research article: Sikorska K., Lesaffre, E., Groenen P.J.F. and Eilers P.H.C. (2013) More GWAS on your notebook: fast mixed models for longitudinal phenotypes using QuickMix. *Manuscript*

5.1 Background

Computations in genome-wide association studies (GWAS) have two aspects. On the one hand vast data are analyzed in search of significant results, making the computations intensive. The amount of data is continuously growing with new imputation strategies as well as analyses of different types of genetic variants. On the other hand, GWA analysis is an "embarrassingly parallel problem" which can be solved by multiprocessor computing. The approaches to those intensive computations are also twofold. Large computing resources can be used to perform the analysis within a reasonable time frame. This is a common solution. The second approach is to develop algorithms that hugely speed up the computations. We, following of the latter route, recently presented "semi-parallel" algorithms for linear and logistic regression, which offer several attractive improvements (Sikorska et al., 2013a). We translated the computations to large matrix operations increasing their speed by several orders of magnitude, making them feasible on a single everyday computer. The name "semi-parallel" indicates that many SNPs are handled at the same time, but within the algorithm and not by using multiple processors. We implemented our approach in just a handful of lines of pure R code, performing exactly linear regression and approximating logistic regression very precisely.

Modeling cross-sectionally measured continuous or binary response covers the majority of present-day GWAS, but gradually the field is being extended to longitudinal phenotypes. One of the advantages of using such data is that they may improve the precision of the estimated mean effects over the single visit analysis (Kerner et al., 2009). However, the main purpose of analyzing longitudinal data is to find genetic variants influencing the evolution of a phenotype over time. The complexity of the computations increases enormously, because a linear regression model is no longer enough. The key points are to take care of the dependence between observations from the same individual, together with irregularity of the measurements' occasions and missing data. Mixed models (MM) constitute one way to solve these issues, at the price of more intensive computations. A typical fit of a MM takes around one second. With a million SNPs this becomes two weeks. In our experience there is a growing amount of data and interest in "longitudinal GWAS", however the computations can be a real hassle or even a limiting factor.

In Sikorska et al. (2013b) we proposed a solution, reducing mixed models to linear regression, after which semi-parallel regression can be applied. This approach is extremely fast, cutting down the computations for one million SNPs to a quarter of an hour. However, this method is providing an approximation of the p-values for only the SNP$\times$time interaction effect in the model.

In this paper we present another approximate method to large-scale fitting of MMs, providing estimated effects and standard errors for both cross-sectional and longitudinal SNP effects. We exploit two characteristics of the problem. Firstly, we write down MM estimation as penalized least squares, where the MM variance components are translated into penalty parameters. Secondly, we assume that the penalty matrix is known (from the model without a SNP) and does not need to be re-estimated for each SNP. This assumption seems to be very reasonable considering large sample sizes and the very small effect sizes characterizing GWAS.

The penalized least squares problem leads to a large system of equations, based on a bordered block-diagonal matrix. We avoid forming this matrix explicitly (not even using sparse matrices), but solve the equations stepwise in a semi-symbolic way. Fast MM computations are more complex than semi-parallel linear regression and also some convenient properties are lost, for example, the possibility to project out the covariates. Therefore, another crucial point is to compute as many components of those equations as possible, before introducing a SNP into the model.

Our implementation is not as concise as in semi-parallel regression, but it is still done in pure R code and it is still very fast. We can handle 5000 individuals with 5 observations each, 5 covariates and one million SNPs within half an hour, again on an average notebook. This motivated us to calling our method *QuickMix* emphasizing the great speed of the computations.

The approximation is almost exact for estimated betas and very precise for the p-values in the range of a typical GWAS, $p > 5 \times 10^{-10}$.

With this speed of the computations, as in semi-parallel regression, data access becomes a bottleneck. In Sikorska et al. (2013a) we discussed how to arrange the storage of SNP data in such a way that blocks of SNPs, containing all subjects, can be accessed quickly. We assume that this data reorganization has been performed. For details we refer to the abovementioned paper.

5.2 Methods

Mixed models as penalized least squares

Let us consider the general formulation of a linear mixed model (Laird and Ware, 1982)

$$\begin{cases} Y_i = X_i\beta + Z_i b_i + \epsilon_i, \quad i = 1, ..., n \\ b_i \sim N(0, D) \\ \epsilon_i \sim N(0, \Sigma_i) \\ b_1, ..., b_n, \epsilon_1, ..., \epsilon_n \quad \text{independent.} \end{cases} \tag{5.1}$$

In (5.1) Y_i is k_i dimensional vector of responses for individual i, X_i and Z_i are $k_i \times p$ and $k_i \times q$ dimensional matrices of covariates, β is a p-dimensional vector of coefficients identical for all individuals and b_i is a q-dimensional vector containing random effects. Measurement error is represented by the k_i-dimensional vector ϵ_i. Furthermore, D is $q \times q$ variance-covariance matrix of random effects and Σ_i is $k_i \times k_i$ variance-covariance matrix of measurement error. In case of balanced data with an equal number of observations per individual we have $k_i = k$. There are several methods for fitting model (5.1) that differently approach the estimation of the random effects. One of the approaches evaluates fixed parameters and variance components using (restricted) maximum likelihood ((RE)ML). Next, since the subject-specific parameters are assumed to be random variables, Bayesian methods are used for their estimation, resulting in "Empirical Bayes" estimates.

Henderson et al. (1959) showed that conditional on the variance components, the fixed and random effects can be estimated jointly by solving the system of equations:

$$\begin{pmatrix} X'\Sigma^{-1}X & X'\Sigma^{-1}Z \\ Z'\Sigma^{-1}X & Z'\Sigma^{-1}Z + \mathcal{D}^{-1} \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X'\Sigma^{-1}y \\ Z'\Sigma^{-1}y \end{pmatrix}, \tag{5.2}$$

where X, y, b are obtained by stacking X_i, y_i, b_i, respectively, underneath each other. Furthermore, $\mathcal{D}, \Sigma$ and Z are block diagonal matrices with D, Σ_i and Z_i on the main diagonal and zeros elsewhere.

It is easier for the future computations to rewrite the joint variance-covariance matrix of random effects and measurement error in a relative variance-covariance form as in Robinson (1991):

$$\text{var} \begin{pmatrix} b \\ \epsilon \end{pmatrix} = \begin{pmatrix} Q & 0 \\ 0 & R \end{pmatrix} \sigma^2, \tag{5.3}$$

where $Q = \mathcal{D}/\sigma^2$. We consider here the model with independent and homoscedastic measurement error, so R simplifies to the identity matrix I. Now system (5.2) can be written as:

$$\begin{pmatrix} X'X & X'Z \\ Z'X & Z'Z + Q^{-1} \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X'y \\ Z'y \end{pmatrix}. \tag{5.4}$$

In Robinson (1991) it has been shown that system (5.4) can be also obtained from minimization of the objective function

$$\|y - X\beta - Zb\| + b'Q^{-1}b$$

with respect to β and b. Hence, we are solving a penalized least squares problem with a penalty imposed on random effects. In the remainder of this article we will replace Q^{-1} with P, denoting the penalty matrix. The connection between MMs and penalized least squares (PLS) has already been made a number of times. In Gurrin et al. (2005) and Wand (2003), this relationship facilitates using the general mixed model software for smoothing. On the other hand, in the R package **lme4**, this representation of a MM speeds up estimation through the exploitation of sparse matrices.

Mixed model in longitudinal GWAS

We consider the following linear mixed model for a continuous phenotype Y measured for individual i at the time t_{ij}

$$Y_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2 g_i + \beta_3 g_i t_{ij} + \sum_{h=1}^{l} \gamma_h c_{hij} + b_{0i} + b_{1i} t_{ij} + \epsilon_{ij}, \tag{5.5}$$

where g_i denotes the genotype for individual i which is either the SNP allele count (integer number) or the dosage (continuous number) and c_{hij} is the value of the h-th additional covariate. Model (5.5) is a so-called "random intercept + slope model", in which b_{0i} and b_{1i} describe individual variation in the intercept and the slope respectively. Depending on the time origin, the intercept usually corresponds to the baseline or average value. It is commonly assumed that b_{0i} and b_{1i} have a multivariate normal distribution with zero means and variance-covariance matrix

$$D = \begin{pmatrix} \sigma_0^2 & \rho\sigma_0\sigma_1 \\ \rho\sigma_0\sigma_1 & \sigma_1^2 \end{pmatrix}. \tag{5.6}$$

Model (5.5) can be fitted in two freely available R packages: **nlme** and **lme4**. Average computation time for one model is 4 and 1 seconds, respectively. That implies at least 12 days of computation for 1 million of SNPs, using the faster package.

Normally, the D matrix is estimated every time for each SNP. It is not unreasonable, however, to assume that it not much different from D^* which is estimated for model (5.5) but without both SNP components. Formal derivation of this concept has already been explored by us in another article (Sikorska et al., 2014). There we have shown that D^* has the form

$$\begin{pmatrix} \sigma_0^2 + \beta_2^2\text{var}(g) & \rho\sigma_1\sigma_2 + \beta_2\beta_3\text{var}(g) \\ \rho\sigma_1\sigma_2 + \beta_2\beta_3\text{var}(g) & \sigma_1^2 + \beta_3^2\text{var}(g) \end{pmatrix}. \tag{5.7}$$

When a SNP is not important in the model, i.e β_2 and β_3 are practically zero, D^* is essentially equal to D. This is the case for most of the SNPs in GWAS. In the situation when SNP has an effect (cross-sectional and/or longitudinal), the variances in D^* will be inflated. The cross-sectional effect inflates the variance of the random intercept, while the longitudinal effect affects the variance of the random slope. The magnitude of this inflation depends on the β_2 and β_3. The covariance in D^* is influenced only if both SNP effects are non-zero.

With respect to the measurement error variance, we can assume that the model without a SNP provides the almost exact value of σ^2. It has been discussed in Orelien and Edwards (2008) that σ^2 is robust against misspecification in the fixed effects if those effects have also corresponding random component in the model. The effect of the inflation in the variance components on the estimates, standard errors and consequently p-values of the SNP effects is explored later in this article and summarized in the Results section.

Once the assumption of known variance components has been made, the computations in *QuickMix* are exact. Below we describe the idea and used computational tricks.

The QuickMix algorithm

No covariates, no SNP

We start with the mixed model in which we describe the evolution of Y as a function of only time given by

$$Y_{ij} = \beta_0 + \beta_1 t_{ij} + b_{0i} + b_{1i} t_{ij} + \epsilon_{ij}. \tag{5.8}$$

We can write the system of equations solving $\beta = (\beta_0, \beta_1)'$ and $b = (b_{01}, b_{11}, b_{02}, b_{12}, ..., b_{0n}, b_{1n})'$ in penalized least squares manner as follows:

$$\begin{pmatrix} T'T & T'Z \\ Z'T & Z'Z + P \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} T'y \\ Z'y \end{pmatrix}. \tag{5.9}$$

In (5.9) T is a fixed design matrix with ones in the first column and time points in the second column and Z is a block diagonal matrix with T_i on the diagonal and zeros elsewhere. Below we write system (5.9) for 3 individuals more explicitly:

$$\left(\begin{array}{c|ccc} \sum_i S_i & S_1 & S_2 & S_3 \\ \hline S_1 & S_1 + P & 0 & 0 \\ S_2 & 0 & S_2 + P & 0 \\ S_3 & 0 & 0 & S_3 + P \end{array}\right) \begin{pmatrix} \beta \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \left(\begin{array}{c} \sum_i r_i \\ \hline r_1 \\ r_2 \\ r_3 \end{array}\right), \tag{5.10}$$

where:

$$S_i = \begin{pmatrix} \sum_j 1 & \sum_j t_{ij} \\ \sum_j t_{ij} & \sum_j t_{ij}^2 \end{pmatrix}, b_i = \begin{pmatrix} b_{0i} \\ b_{1i} \end{pmatrix} \quad \text{and} \quad r_i = \begin{pmatrix} \sum_j y_{ij} \\ \sum_j t_{ij} y_{ij} \end{pmatrix}.$$

The above system has a block structure

$$\begin{pmatrix} A_{11} & A_{21}' \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}, \tag{5.11}$$

with the explicit solution given by:

$$\beta = (A_{11} - A_{21}' A_{22}^{-1} A_{21})^{-1} (q_1 - A_{21}' A_{22}^{-1} q_2) \quad \text{and} \quad b = A_{22}^{-1}(q_2 - A_{21}\beta). \tag{5.12}$$

In (5.12) A_{22} is a block diagonal matrix with 2×2 matrices $S_i + P$ on its diagonal. It is convenient for the further computations to transform it into an identity matrix, so that the inverse of this matrix does not need to be computed. It is achieved in two steps using singular value decomposition (SVD) as follows. We decompose each $S_i + P$ into $U_i \Omega_i V_i'$, such that $U_i' U_i = I, V_i' V_i = I$ and Ω_i is a diagonal matrix. Matrix $S_i + P$ is symmetric and therefore its singular value decomposition simplifies to $U_i \Omega_i U_i$. Now, we can write the second equation in (5.10) as

$$S_i \beta + U_i \Omega_i U_i b_i = r_i. \tag{5.13}$$

Substituting $U_i c_i$ for b_i and premultiplying both sides of the equation by U_i gives us:

$$U_i S_i \beta + \Omega_i c_i = U_i r_i, \tag{5.14}$$

which transforms matrix A_{22} from block diagonal to diagonal. In the second step we replace c_i with $\Omega^{-1/2}\theta_i$ and premultiply both sides of the equation by $\Omega^{-1/2}$ which leads to a new system of equations given by

$$\left(\begin{array}{c|ccc} \sum_i S_i & (\Phi_1 S_1)' & (\Phi_2 S_2)' & (\Phi_3 S_3)' \\ \hline \Phi_1 S_1 & I & 0 & 0 \\ \Phi_2 S_2 & 0 & I & 0 \\ \Phi_3 S_3 & 0 & 0 & I \end{array}\right) \begin{pmatrix} \beta \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{pmatrix} = \left(\begin{array}{c} \sum_i r_i \\ \hline \Phi_1 r_1 \\ \Phi_2 r_2 \\ \Phi_3 r_3 \end{array}\right), \tag{5.15}$$

where Φ_i is a rotation matrix for individual i defined as $\Omega_i^{-1/2}U_i$. This system has the block structure

$$\begin{pmatrix} A_{11} & A_{21}^{*\prime} \\ A_{21}^{*} & I \end{pmatrix} \begin{pmatrix} \beta \\ \theta \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2^{*} \end{pmatrix}. \tag{5.16}$$

Now the system of equations in (5.12) is simplified to

$$\beta = (A_{11} - A_{21}^{*\prime}A_{21}^{*})^{-1}(q_1 - A_{21}^{*\prime}q_2^{*}) \quad \text{and} \quad \theta = q_2^{*} - A_{21}^{*}\beta. \tag{5.17}$$

The part of the equations related to the fixed effects remains unchanged. The random effects have been transformed, but they are not of interest in GWAS. Moreover, one can always transform the θ back into b.

Adding covariates

If C denotes a $\Sigma_i k_i \times l$ dimensional matrix of covariates, the solution of the extended model is given by system (5.4) with $X = (T \quad C)$. In semi-parallel regression covariates were very easy to handle via projections. This trick is unfortunately not possible in mixed models framework and we need to solve the system for $l + 2$ parameters. The system for the model with covariates is extended in 3 out of 4 blocks in the left-hand side, leaving the bottom-right block diagonal structure unchanged. Therefore, it is again convenient to apply rotation bringing this matrix to identity. After applying the SVD transformation we can write the system for 3 individuals as follows

$$\left(\begin{array}{c|ccc} X'X & (\Phi_1 T_1' X_1)' & (\Phi_2 T_2' X_2)' & (\Phi_3 T_3' X_3)' \\ \hline \Phi_1 T_1' X_1 & I & 0 & 0 \\ \Phi_2 T_2' X_2 & 0 & I & 0 \\ \Phi_3 T_3' X_3 & 0 & 0 & I \end{array}\right) \begin{pmatrix} \alpha \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{pmatrix} = \begin{pmatrix} X'y \\ \Phi_1 T_1' y_1 \\ \Phi_2 T_2' y_2 \\ \Phi_3 T_3' y_3 \end{pmatrix}, \tag{5.18}$$

where $\alpha = (\beta \quad \gamma)'$. Now the solution is equal to (5.17) with the corresponding blocks coming from (5.18).

Solving SNP effects

We add a SNP to the model creating a border to the previous system of equations. Two effects, cross-sectional and longitudinal are added, so G is a $\Sigma_i k_i \times 2$ dimensional matrix. We again divide this system for blocks but this time we separate the SNP border instead of dividing fixed and random parts

$$\left(\begin{array}{c|cc} G'G & G'X & G'Z \\ \hline X'G & X'X & X'Z \\ Z'G & Z'X & Z'Z + P \end{array}\right) \begin{pmatrix} \xi \\ \alpha \\ b \end{pmatrix} = \begin{pmatrix} G'y \\ X'y \\ Z'y \end{pmatrix}, \tag{5.19}$$

where $\xi = (\beta_2, \beta_3)'$. We can write system (5.19) after SVD transformation in a block form

$$\begin{pmatrix} H_{11} & H_{21}' \\ H_{21} & H_{22} \end{pmatrix} \begin{pmatrix} \xi \\ \psi \end{pmatrix} = \begin{pmatrix} J_{11} \\ J_{21} \end{pmatrix}, \tag{5.20}$$

with

$$H_{21} = \begin{pmatrix} X'G \\ \Phi_1 T_1' G_1 \\ \Phi_2 T_2' G_2 \\ \vdots \\ \Phi_n T_n' G_n \end{pmatrix}, \quad J_{21} = \begin{pmatrix} X'y \\ \Phi_1 r_1 \\ \Phi_2 r_2 \\ \vdots \\ \Phi_n r_n \end{pmatrix}.$$

Solving system (5.20) for ξ gives us

$$\xi = (H_{11} - H_{21}' H_{22}^{-1} H_{21})^{-1} (J_{11} - H_{21}' H_{22}^{-1} J_{21}). \tag{5.21}$$

Clearly, in (5.21) $H_{22}^{-1} J_{21}$ and $H_{22}^{-1} H_{21}$ are the most expensive operations, since they involve inverting $(2n + p + 2) \times (2n + p + 2)$ dimensional matrix. However, in none of those matrices H_{22} needs to be solved explicitly. Note that $H_{22}^{-1} J_{21}$ is a vector with solutions of system (5.18) which after being solved once is stored and treated as constant in the further computations. The second operation is a solution for the mixed model given in (5.18) but for a different right hand side, namely H_{21}. Note that in this case the RHS of the system is two-dimensional.

We showed here how the genetic effects can be estimated efficiently, using the idea of bordered matrices and exploiting elements computed for the model without the SNP.

Standard errors

To compute the variance-covariance matrix of the estimated fixed and random effects in a MM we need to invert the LHS matrix of system (5.2) (Ruppert et al., 2003). Standard errors are equal to the square roots of the diagonal elements of that matrix. In penalized least squares (PLS) notation, we need to invert LHS of system (5.4) and multiply diagonal elements by σ^2. In our case we are interested only in the inference for SNP effects. They are the upper-left part of the expression

$$\sigma^2 \begin{pmatrix} H_{11} & H_{21}' \\ H_{21} & H_{22} \end{pmatrix}^{-1}. \tag{5.22}$$

Using the formula for the matrix inverse in block form, the standard errors of ξ are given by

$$\sigma \sqrt{(\text{diag}(H_{11} - H_{21}' H_{22}^{-1} H_{21})^{-1})}. \tag{5.23}$$

Note that this diagonal has already been computed in (5.21) showing that the computation of the standard errors is trivial.

Missing data

Typically in longitudinal studies we deal with data sets in which individuals have different number of available outcome measurements. It may also be the case that the outcome value is available but at least one of the covariates is missing. If a covariate is related to a baseline characteristic, the whole individual profile is removed from the analysis. However, if the the covariate is time-varying, the available measurements still contribute to the estimation.

This is one of the advantages of the mixed model analysis that fully exploits gathered data. We assume that the user's data are arranged in a long-format also used by the R packages **lme4** and **nlme** and that there is an equal number of rows per individual, namely k, with missing data coded as "NA". To apply *QuickMix* all "NA's" have to be removed from the database. We can replace them with any numeric value and introduce weighting matrix with zeros and ones on the diagonal indicating if the observation if valid or not. The PLS system of mixed model equations looks then as

$$\begin{pmatrix} X'WX & X'WZ \\ Z'WX & Z'WZ + P \end{pmatrix} \begin{pmatrix} \beta \\ b \end{pmatrix} = \begin{pmatrix} X'Wy \\ Z'Wy \end{pmatrix}. \tag{5.24}$$

One then has to adjust the solution (5.21) for the weights leading to system

$$\begin{pmatrix} H_{w11} & H'_{w21} \\ H_{w21} & H_{w22} \end{pmatrix} \begin{pmatrix} \xi \\ \psi \end{pmatrix} = \begin{pmatrix} J_{w11} \\ J_{w21} \end{pmatrix}, \tag{5.25}$$

noting that SVD is now applied to a matrix $S_{wi} + P$, where

$$S_{wi} = \begin{pmatrix} \sum_i w_i & \sum_i w_i t_i \\ \sum_i w_i t_i & \sum_i w_i t_i^2 \end{pmatrix}.$$

The dimension of matrix W is usually large, $nk \times nk$. However, this matrix is never formed explicitly and only its diagonal is stored as a vector.

Formally, one should take the proper degrees of freedom (excluding observations with weight 0) in the calculation of p-values. It has no influence on our computations as we use the normal approximation of the distribution of the t-statistics. Note also that the approach using weights is technically equivalent to replacing the whole missing row with zeros and skipping the weighting matrix. To achieve a higher speedup we use this approach in our computations. Regarding SNPs, as in our previous work, we assume that genotype data after imputation is complete and otherwise imputed with the sample mean.

Implementation

QuickMix is written purely in R, but the code is much more lengthy than for semi-parallel regression. Therefore, we provide only the pseudo-code here and the R implementation can be found in the Appendix.

The SNP data is stored in a $n \times m$ dimensional matrix, where m is the number of SNPs. In order to fit the mixed model with for example **lme4**, a SNP vector of length n has to be extended to the length nk. This is time consuming for a large n. We avoid it in our algorithm, knowing that a SNP is constant over time. Additionally it is useful to notice that

$$G_i = \begin{pmatrix} g_i & g_i t_{i1} \\ g_i & g_i t_{i2} \\ \vdots & \\ g_i & g_i t_{ik} \end{pmatrix} = g_i T_i. \tag{5.26}$$

Our algorithm has four phases. The first one is data preparation and getting the penalty matrix which is later treated as known through the remaining computations. Second part consists of solving efficiently the model without the SNP and storing key elements. In the third phase we precompute some elements in the SNP border of system (5.20). Using the fact that genotype is constant over time we can store individual sum of products $\sum_j w_{ij}x_{ij}$ for all X columns. This speeds up the computations during the estimation of the genetic effects and additionally keeps the SNP data as compact as possible. Finally we loop over SNPs estimating their cross-sectional and longitudinal effects. Below we summarize the computations in pseudo-code.

1. Read phenotype data.
2. Fit the mixed model without SNP with $lmer/lme$.
3. Extract the variance-covariance matrix and compute the penalty matrix.
4. Compute the weights for missing data and store them in a vector.
5. Replace rows with missing values by rows with all values being zero.
6. Build matrices T, X and Y needed to solve system (5.18).
7. In a loop for i to n compute SVD$(S_i + P)$ and build matrices A_{21}^* and q_2^*.
8. Compute and store the solution given in (5.17).
9. Compute $\sum_j w_{ij}x_{ij}$ for all individuals using row-wise Kronecker products.
10. Read SNP data.
11. Loop over SNPs.
    a) Compute solution given in (5.21).
    b) Compute standard errors given in (5.23).

In GWA analysis not all SNP data fit into software memory. This has already been discussed in our previous work (Sikorska et al., 2013a). We advertise using array-oriented binary files implemented in a number of R packages, e.g. **ncdf** and **ff**. Likewise semi-parallel regression, *QuickMix* works with blocks of SNPs repeating step 11 over the blocks.

5.3 Results

Simulation study

For evaluating the performance of *QuickMix*, we are interested in the magnitude of the computational speedup and the accuracy of the estimation. Our algorithm uses only one approximation: it assumes that the penalty matrix does not change when a SNP is introduced. For the rest, the computations are exact.

Theoretical justification of the assumption about known penalty matrix has been discussed in the Methods section where we compared D with D^*. Practically, the magnitude

of the misspecification in D^* depends on the values of β_2 and β_3. In cross-sectional GWAS framework, the SNP effects contribute to the total variance explained by the model, measured by R^2, by no more than 1%. However, unlike in linear regression, the concept of variance explained in mixed models is much more complicated and not fully resolved. In Orelien and Edwards (2008) it has been shown that the available measures of variance explained for mixed model framework are inappropriate for evaluating the contribution of the fixed predictors on the individual level (conditional R^2). By saying inappropriate we mean that they do not necessarily increase once the significant predictors are added to the model or that they can even have negative values. On the other hand, the marginal R^2 seems to be a good measure (with desirable properties) of the goodness of fit. Several choices of the marginal R^2 exist in the published literature, e.g Vonesh et al. (1996), Vonesh and Chinchilli (1997) and Xu (2003). Because our aim is only to get an idea of a maximum SNP effect that we can expect in GWAS, we chose the measure of variance explained by fixed effects, which is the most intuitive and the simplest in implementation, provided in Xu (2003):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)'(y_i - \hat{y}_i)}{\sum_{i=1}^{n}(y_i - \bar{y}\mathbf{1}_{n_i})'(y_i - \bar{y}\mathbf{1}_{n_i})}, \tag{5.27}$$

where $\hat{y}_i = X_i\hat{\beta}$.

We explored through simulations the influence of the misspecification in the matrix P on the estimates, standard errors and p-values of the SNP effects. We also computed R^2 for each simulation to make sure that the chosen values for β_2 and β_3 are reasonable within the GWAS framework.

Results of the simulation study

The estimates are basically exact through the whole scale of their values (Figure 5.1). The standard errors of *QuickMix* are somewhat overestimated for the larger values of β's, which is expected as variances in P are inflated due to omitted SNP effects (Figure 5.2). However, our main interest lies in p-values. Due to overestimated standard errors, the $-\log 10(p)$ for larger betas are too pessimistic. Nevertheless, they increase monotonically with larger effect sizes, just with bias downward with respect to the $-\log 10(p)$ from *lmer*. Moreover, Figure 5.3 shows that the p-values obtained in *QuickMix* are almost exact in the significance range common in GWAS (up to 10 on $-\log 10$ scale), which guarantees the correct probability of type I error. To generate those results we used simulated data for 2000 individuals and the maximum contribution to R^2 of the SNP effects around 6%.

We explored the speedup of *QuickMix* with respect to *lmer* depending on the sample size. We are also interested in the computation time of our method for 1 million of SNPs, assuming 5 observations per individual and 5 additional covariates. We performed the computations on a single desktop computer with Intel i5-3470, 8 GB of RAM and 64 bit version of R. The results are displayed in Figure 5.4. The achieved speedup is substantial. *QuickMix* performs around 1000 times faster than *lmer*. We notice that in general the benefit become larger when the sample size increases. A GWA on 1 million SNPs can be done within half an hour for around 10K of individuals.

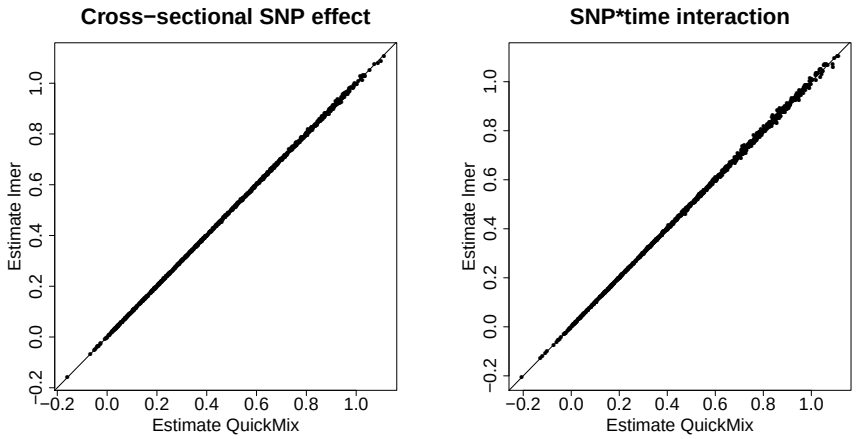


Figure 5.1: Precision of the estimates obtained by *QuickMix*.

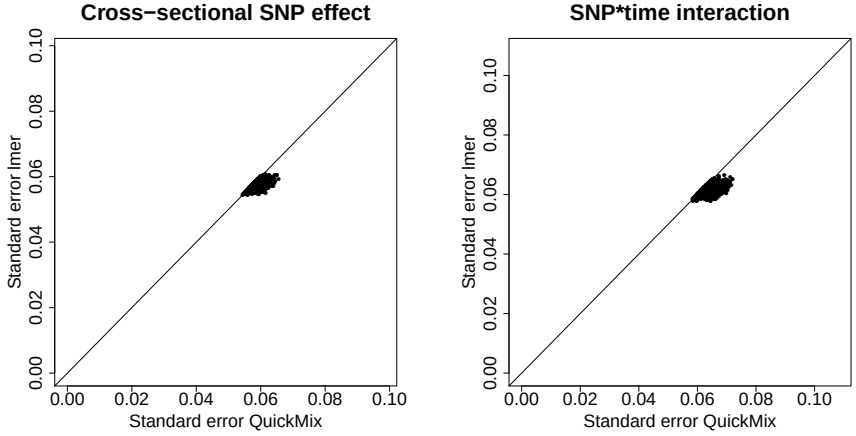


Figure 5.2: Precision of the standard errors obtained by *QuickMix*.

5.4 Conclusions

We presented *QuickMix*, an algorithm which speeds up a GWA scan for 1 million SNPs from two weeks to half an hour on an everyday computer. The key idea is to write the mixed model equations as penalized least squares. Additionally *QuickMix* relies on only one, very reasonable, assumption about the penalty matrix, which has been justified here analytically and by simple simulations. Apart from this assumption, the remaining computations are exact. This is an attractive property, leading to very precise computation of the p-values. The implementation is purely based on R functions, using basic matrix operations. It is important to keep computations and data as compact as possible and therefore we use

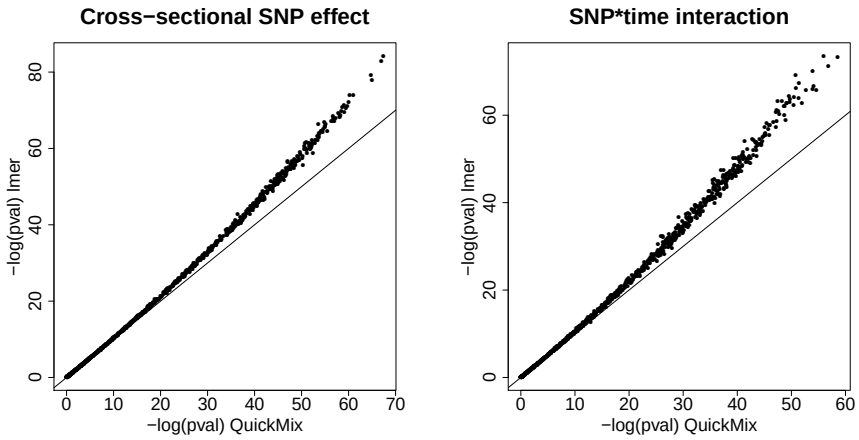


Figure 5.3: Precision of the p-values obtained by *QuickMix*.

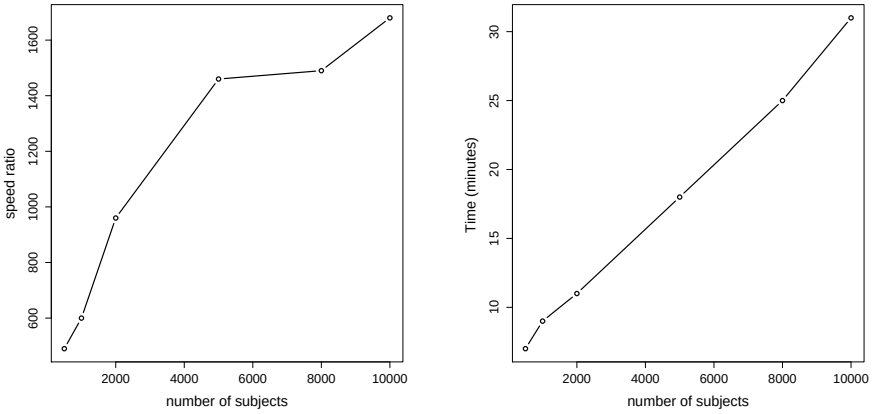


Figure 5.4: On the left panel speedup achieved by *QuickMix* depending on sample size. On the right panel computation time for 1 M of SNPs using *QuickMix* versus sample size.

many computational tricks storing key elements before looping over SNPs. We provide the R code as supplementary material allowing readers an easy start with *QuickMix*.

As in the semi-parallel approach, with a fast speed of computations data access becomes a bottleneck. The solution to that, presented in our previous manuscript (Sikorska et al., 2013a), also applies here. Using packages **ff** or **ncdf** we can read the genotype data for 1 million of SNPs within 10-15 minutes. A combination of data organization based on array-oriented binary files together with *QuickMix* makes it feasible to perform a longitudinal GWAS for several millions of SNPs on a single laptop within just a few hours.

However, there are still more challenges waiting in this area. One could think of a longitudinal data gathered within families, where many sources of correlation need to be modeled in the same time, or nonlinear trends involving polynomials or even splines. In a GWAS framework all these types of modeling need a great speedup to facilitate a reasonable time of analysis.

Appendix

The R code provided below solves the mixed model for ns simulated SNPs using *QuickMix*. We assume that the phenotype data denoted in the code as `Data` a have long structure, i.e repeated observations on the same individual are in separate rows. Additionally the data frame has nk rows and missing data are coded as "NA".

```
# Fit model without SNP using lmer
# covs are names of all additional covariates in the model
mod_formula <- as.formula(paste("y ~ time +",
 paste(covs, collapse= "+"), "+ (time|id)"))
mod1 = lmer(mod_formula, data = Data, na.action = "na.omit")

# Extract variance components and compute penalty matrix (P)
varcor = VarCorr(mod1)
G = varcor$id
sig = attr(varcor, "sc")
P = solve(G / sig ^ 2)
# Put data in convenient arrays and vector
TT = XY[ , 1 : 2]
nxy = ncol(XY)
X = XY[ , -nxy]
nx = ncol(X)
y = XY[ , nxy]

# Compute components of block-diagonal
# system with covariates  (without SNP)
# Additionally compute and store object SS = Rot %*% Si
#which is used later on
A21 = matrix(NA, 2 * n , 2 + ncov)
q2  = rep(NA, 2 * n)
SS = matrix(NA, 2 * n , 2)
for (i in 1 : n ) {
  uk = (i - 1) * k + (1 : k)
  u2 = (i - 1) * 2 + (1 : 2)
  Ti = TT[uk, ]
  Si = crossprod(Ti, Ti)
  sv = svd(Si + P)
  Rot = sqrt(1 /sv$d) * sv$u
  Q = Rot %*% t(Ti)
  SS[u2, ] = Rot %*% Si
  A21[u2, ] = Q %*% X[uk,  ]
  q2[u2] = Q %*% y[uk]
}
q1 = crossprod(X, y)
A11 = crossprod(X)
# Solve the system (20)
```

```
Q = A11 - crossprod(A21)
q = q1 - crossprod(A21, q2)
sol = solve(Q, q)
blups = q2 - A21 %*% sol

# Compute sums of products per subject involved in the
# crossprod(X, G), crossprod(G) and crossprod(G, y)
# We use row-wise Kronecker product to avoid repeating
# SNP vector k times
ex = matrix(1, 1, ncov + 2)
et = matrix(1, 1, 2)
XTk = kronecker(et, X) * kronecker(TT, ex)
TTk = kronecker(et, TT) * kronecker(TT, et)
Tyk = y * TT
XTs = matrix(0, n, ncol(XTk))
TTs = matrix(0, n, ncol(TTk))
Tys = matrix(0, n, 2)
AtS = matrix(0, n, 2 * nx)
for (i in 1:n) {
  uk = (i - 1) * k + (1 : k)
  XTs[i, ] = apply(XTk[uk, ], 2, sum)
  TTs[i, ] = apply(TTk[uk, ], 2, sum)
  Tys[i, ] = apply(Tyk[uk, ], 2, sum)
  u2 = (i - 1) * 2 + (1 : 2)
  AtS[i, ] = c(crossprod(A21[u2, ], SS[u2, ]))
}
# Add SNPs one by one and solve
Theta = D = matrix(NA, ns, 2)
for (i in 1 : ns) {
  si = SNPS[, i]
  snp2 = rep(si, each = 2)
  H1 = matrix(crossprod(si, XTs), nx, 2)
  H2 = snp2 * SS
  AtH = matrix(crossprod(si, AtS), nx, 2)
  R = H1 - AtH
  Cfix = solve(Q, R)
  Cran = H2 - A21 %*% Cfix
  GtG = matrix(crossprod(si ^ 2, TTs), 2, 2)
  Gty = matrix(crossprod(si, Tys), 2, 1)
  V = GtG - crossprod(H1, Cfix) - crossprod(H2, Cran)
  v = Gty - crossprod(H1, sol) - crossprod(H2, blups)
  Theta[i , ] = solve(V, v)
  D[i, ] = diag(solve(V))
}
SE = sqrt(sig2) * sqrt(D)
Pval = 2 * pnorm(-abs(Theta / SE))
```

Chapter 6

GWAS of longitudinal BMD data with 30 million imputed SNPs

Abstract

We report on a large-scale genome-wide association analysis with longitudinal data. The study involved 5000 individuals with 4 measurements of bone mineral density collected over a period of 12 years. The number of SNPs, imputed using the 1000 Genomes Project, is around 30 million. Using the traditional linear mixed model software it would take at least one year to do the computations. We perform the analysis of the BMD data using two approaches: the conditional two-step and Quickmix, speeding up the computations to 4 and 21 hours, respectively. We emphasize strong points of each of the approaches. Many practical issues are being discussed, such as: data access, dealing with large outputs, and plotting. Although computational aspects are our priority, we also report briefly on our findings related to the genetics variants influencing age related change in bone mineral density.

6.1 Phenotype data

The longitudinal data come from the Rotterdam Study (see **Chapter 1**). Around 700 individuals participating in the study did not have any available measurement. It was planned to measure individuals at the baseline and after 2, 6, and 12 years, but the collected data are highly unbalanced. As is typical for large cohort studies, the participants were not measured at the same time points and many of them either dropped out from the study or missed intermittent visits. The number of available BMD measurements versus sample size is described in Table 6.1.

Table 6.1: Number of available BMD measurements (k) versus number of individuals (n) for the Rotterdam Study BMD data.

k	n
1	1107
2	1528
3	1649
4	1281

Due to large gender-related differences in BMD, it is useful to explore the data for men and women separately. The average BMD at baseline was 0.83 $\mathrm{g/cm^2}$ for women and 0.92 $\mathrm{g/cm^2}$ for men. No appreciable change over time was observed. The profile plots are displayed in Figure 6.1

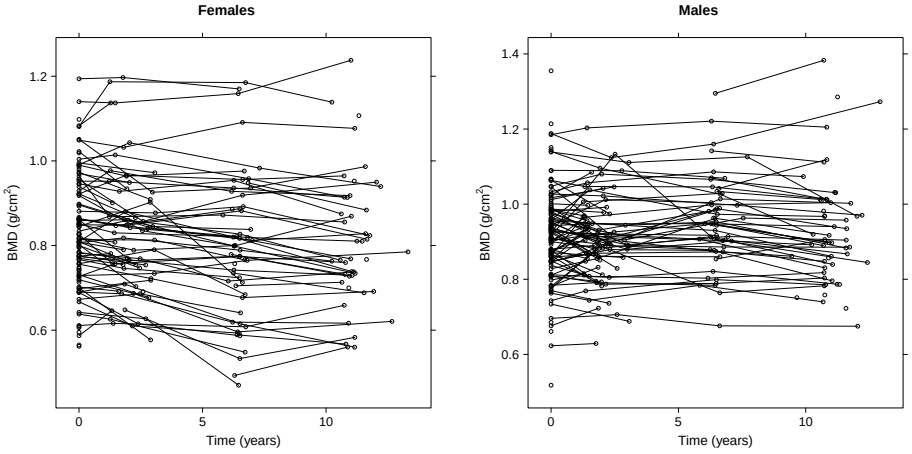


Figure 6.1: Rotterdam Study BMD data. Profile plots for sample of individuals, separately for females and males.

In the BMD data 402 individuals missed their first measurement, but at least one of the subsequent observations was available. In order to increase the statistical power we have treated the first available observation as baseline.

We first build an epidemiological model for the evolution of the BMD omitting genetic effects. The mixed effects model describing the BMD for individual i at time occasion t_{ij} has the form

$$\text{BMD}_{ij} = \beta_0 + \beta_1 t_{ij} + \beta_2 g_i + \beta_3 w_{ij} + \beta_4 a_i + \beta_5 g_i t_{ij} + b_{0i} + b_{1i} t_{ij} + \epsilon_{ij}, \qquad (6.1)$$

where g_i denotes gender for the individual i, a_i denotes the age at baseline, w_{ij} is the body weight at the time t_{ij}. We consider a linear evolution of the BMD over time. Subject-specific coefficients consist of random intercepts (b_{0i}) and random slopes (b_{1i}). Model (6.1) can be easily fitted in R using for example the packages **nlme** or **lme4**. The REML estimates for the parameters are given in Table 6.2.

Table 6.2: REML estimates for fixed effects and variance components for model (6.1).

Parameter	Estimate	Stderr	Pval
β_0	0.986	0.0170	< 0.001
β_1	-0.002	0.0002	< 0.001
β_2	-0.059	0.0034	< 0.001
β_3	0.003	0.0001	< 0.001
β_4	-0.005	0.0002	< 0.001
β_5	-0.002	0.0002	< 0.001
sd(b_{0i})	0.120		
sd(b_{1i})	0.004		
cor(b_{0i}, b_{1i})	-0.057		
sd(ϵ_{ij})	0.038		

Two types of residuals are defined for mixed effects models. The marginal residuals r_m are computed as $Y - X\hat{\beta}$, where Y are the observed outcome values, X is a design matrix for the fixed effects and $\hat{\beta}$ are the (RE)ML estimates for the fixed effects. On the other hand, the conditional residuals r_c are computed as $r_m - Z\hat{b}$, where $\hat{b}$ are BLUPs for the random effects. The marginal and the conditional residuals from model (6.1) are shown in Figures 6.2 and 6.3.

No violations of the normal distribution are observed for marginal residuals (apart from a few outliers). The distribution of the conditional residuals is symmetric but with high kurtosis. The fact that the marginal residuals are normally distributed can be explained by the fact that the normally distributed random intercepts dominate the variability of the marginal residuals (see Table 6.2). We proceed with the untransformed outcomes relying on the fact that the estimation in the linear mixed model for moderate sample sizes is robust against misspecification in the error distributions, as shown in Jacqmin-Gadda et al. (2007).

6.2 Genotype data

The SNP data come from the 1000 Genomes-based imputations. The number of SNPs is around 29 million. As already discussed in the previous chapters, the organization of the

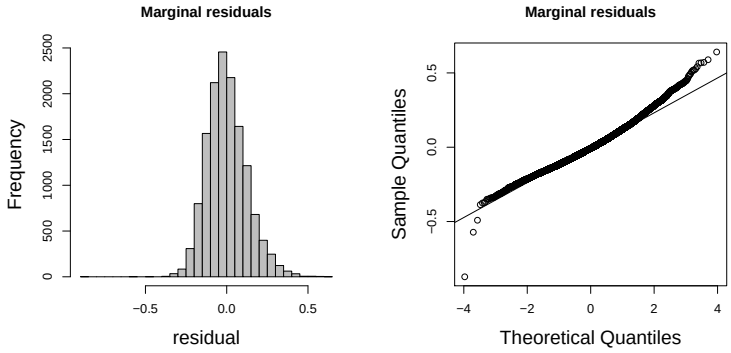


Figure 6.2: Marginal residuals from model (6.1).

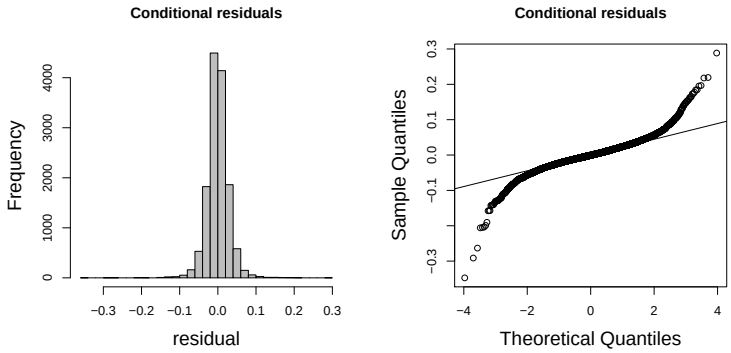


Figure 6.3: Conditional residuals from model (6.1).

genotype data is a vital element of efficient GWAS computations. In **Chapter 4** we argued that the "row per SNP" structure facilitates easy access to blocks of SNPs, unlike the "row per subject" structure. We advised to use array-oriented binary files, where the access to the blocks of SNPs is much more efficient than with MACH/Minimac ASCII files. The data provided to us by our institution used the **DatABEL** format, which is one of the choices for array-oriented binary formats. The SNP data were divided into 198 files, with sizes ranging from 950 MB to 5.4 GB. Additionally, ASCII "info files" store information about reference allele and minor allele frequency (MAF), defined as min(1 - mean(SNP), mean(SNP)). However, MAFs need to be recalculated excluding those 700 individuals with all BMD measurements missing. Finally, so called legend files include the position of the SNP which are needed for Manhattan plotting. Eventually, all those files need to be merged in order to properly summarize the results from the GWA analysis. The total size of the **DatABEL** files is around 700 GB.

6.3 GWA analysis

The goal is to relate the 29 million SNPs to the evolution of BMD over time. The epidemiological model (6.1) is now extended by **SNP** and **SNP**$\times$ **time** effects. The main focus lies on the interaction effect. However, the cross-sectional effect is also of interest. Kerner et al. (2009) discusses that longitudinal study may increase the power of identifying cross-sectionally important variants.

The GWA scan for 29 million SNPs involves fitting 29 million mixed models. This leads to exhaustive computations: 1-2 years of CPU time. We apply 1) the conditional two-step approach (CTS) and 2) the QuickMix algorithm to reduce that time substantially. We combine the conditional two-step algorithm with semi-parallel regression applied to the second step (see **Chapter 4**). By doing this we achieve an additional speedup. Both methods provide the estimates for longitudinal SNP effect, but the main effect can be computed only by using QuickMix.

6.4 Preliminary checks

We performed a preliminary check on one file from the genotype data (a chunk of chromosome 1). We analyzed that data chunk using QuickMix, loading blocks of 10000 SNPs at a time. Next, we randomly selected 5000 SNPs from the whole file and analyzed them using the *lmer* function in R. First we performed the analysis on a single computer with Windows OS and noticed a strange behavior of the method. Around the straight line describing relationship between the p-values from QuickMix and the *lmer*, a few outlying p-values were observed (Figure 6.4). After detailed research on that issue, we concluded that those outlying p-values are not related to the QuickMix algorithm, but to the SNP selection from the **DatABEL** file. Namely, different SNP values were selected during the block selection and during the random SNP selection. To confirm this, we performed exactly the same analysis on a PC with Linux OS and all the outliers vanished (Figure 6.4). We concluded that the problem of the wrong SNP selection is characteristic of the combination of large **DatABEL** files ($>$4 GB) with Windows, suggesting a bug in **DatABEL**. That issue is still being investigated by us and by the developers of the **DatABEL** package.

6.5 Saving the output

Storing and presenting the output becomes an important issue in GWAS with very many SNPs. The number of rows in the output is around 29 million, which results in large ASCII files (several GB). Reading this file with R is either impossible (for a PC with 4-6 GB RAM) or very time consuming. Loading a GWA output saved as a text file took us about 1 hour on a PC with 144 GB RAM. One should keep in mind that the output from the analysis has to be merged with the info and legend files causing an additional challenge. We used

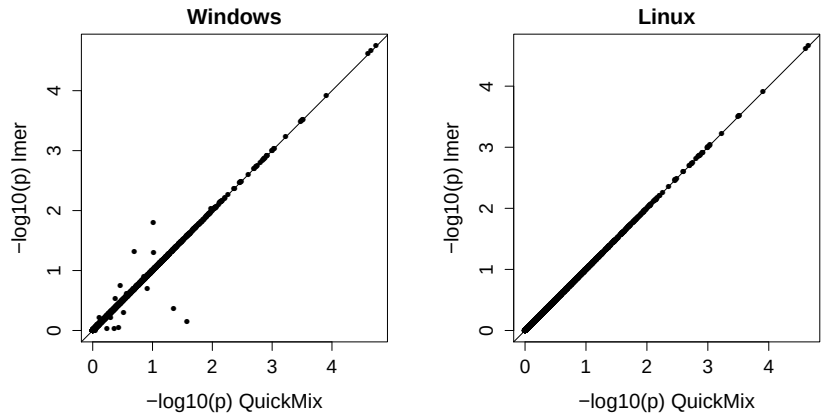


Figure 6.4: Comparison of the p-values between QuickMix and $lmer$ for a large DatABEL file performed under Windows and Linux.

a database that stores all data in separate tables. R software offers the package **RSQLite** which is an interface to SQLite database.

6.6 Results

We performed the analysis on one PC with an Intel X5680 processor running at 3.33 GHz, 144 GB RAM and Linux OS. The analysis with QuickMix took around 21 hours compared to the 4 hours for the conditional two-step approach. Although 29 million SNPs were analyzed, only about 30% (8816273 SNPs) of them survived the filtering with respect to the Minor Allele Frequency (MAF > 0.01) and the quality of imputation (Rsq > 0.3).

We performed two accuracy evaluations, analyzing a sample of SNPs with the *lmer* function of the R package **lme4**. During the first check we reanalyzed 10000 randomly selected SNPs out of all 29 million. To avoid the time-consuming SNP selection we limited ourselves to one of the 198 genotype files. The comparison is shown in Figures 6.5 and 6.6. We noticed that once the very rare SNPs (MAF < 0.01) are excluded, the conditional two-step approach dramatically gains accuracy.

It is important to mention that the conditional two-step approach in fact approximates the conditional linear mixed model. If the latter one would be taken as a reference model, the approximation is almost perfect, as shown in Figure 6.8. In **Chapter 3** we discussed that it may be desirable to take the conditional model as a reference, since it protects us better from any kind of misspecifications in the baseline part of the model.

On the other hand, the main interest lies in the SNPs with low p-values. During the second check, after filtering with respect to MAF > 0.01, we selected the SNPs with a p-value $< 5 * 10^{-4}$. QuickMix selected 5426 SNPs, whereas CTS selected 5216 SNPs. Furthermore, 3995 SNPs were selected with both methods. We reanalyzed the SNPs selected by QuickMix with the function *lmer*. The accuracy is displayed in Figure 6.7.

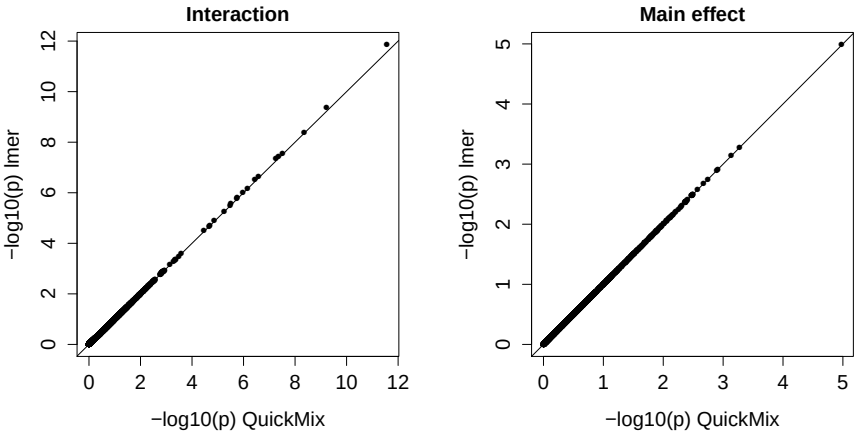


Figure 6.5: Comparison between QuickMix and *lmer* for random SNPs

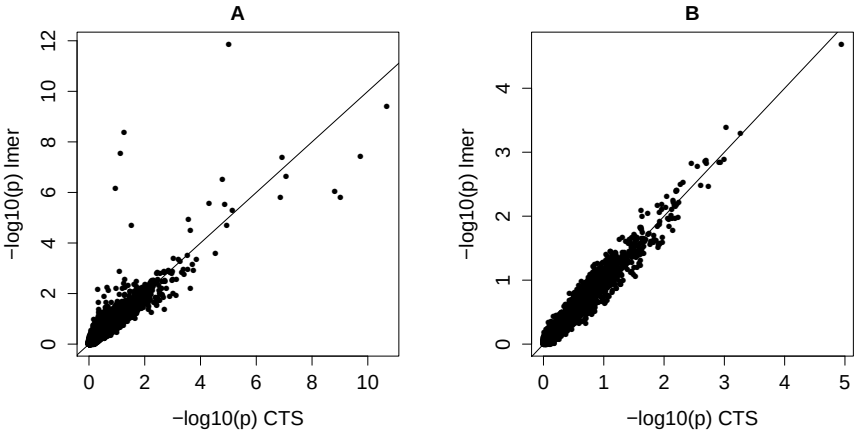
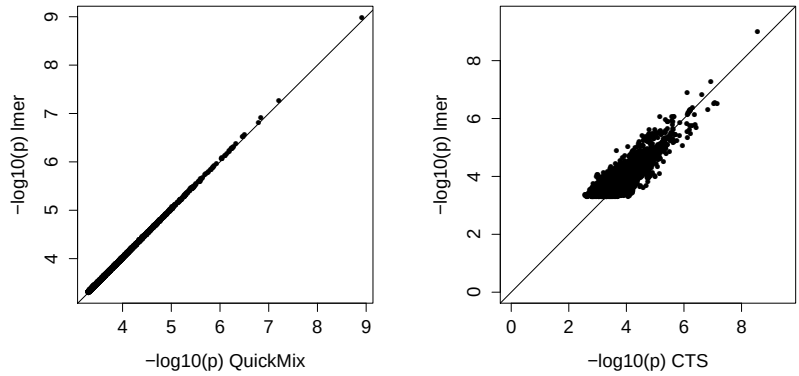


Figure 6.6: Comparison between conditional two-step and *lmer* for random SNPs. In A all the SNPs are present, in B only SNPs with MAF > 0.01.



(a) (b)

Figure 6.7: Comparison between the linear mixed model with the QuickMix (a) and the CTS (b) for the top SNPs ($p < 5 * 10^{-4}$), MAF > 0.01.

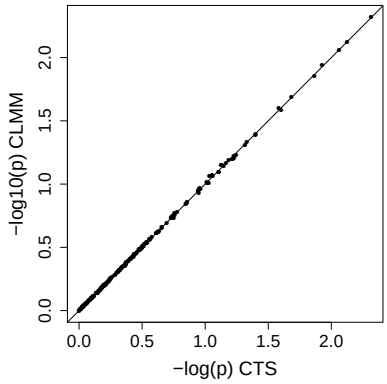
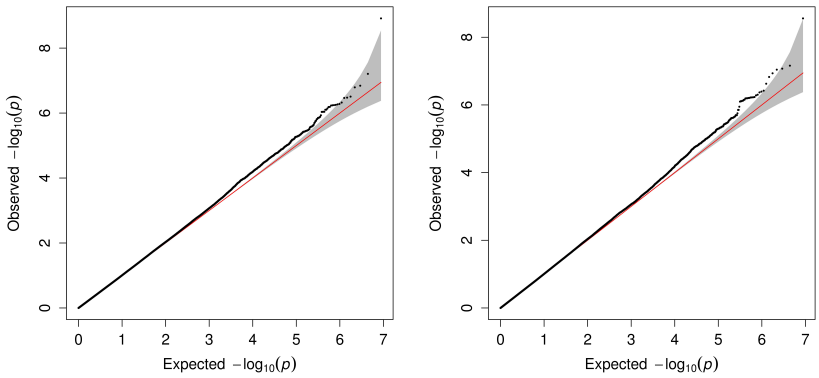


Figure 6.8: Comparison of the p-values from the conditional two-step approach with the p-values from the conditional linear mixed model.

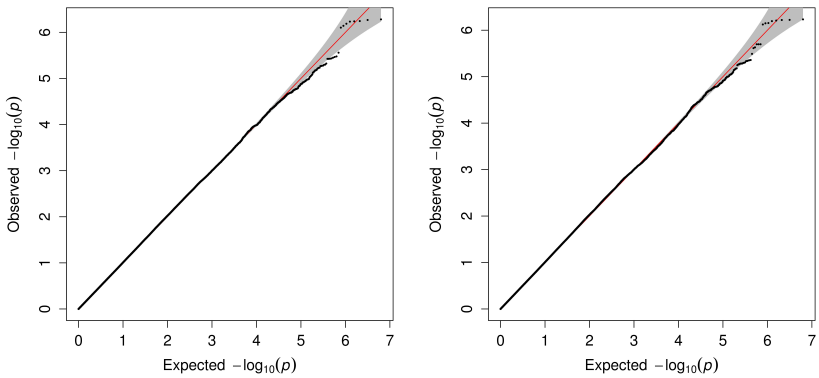
Q-Q and Manhattan plots

Results from GWA analyses are typically summarized using Q-Q and Manhattan plots. They have been described in detail in **Chapter 2**. We plotted the results from QuickMix and CTS for MAF > 0.01 and MAF > 0.05 (Figures 6.9, 6.10, 6.11, 6.12). One SNP reaching the GWA significance level was identified for MAF > 0.01. However, for MAF > 0.05 no SNPs were identified with $p < 5*10^{-8}$. Plots based on the QuickMix and the conditional two-step are very similar, especially for MAF > 0.05.



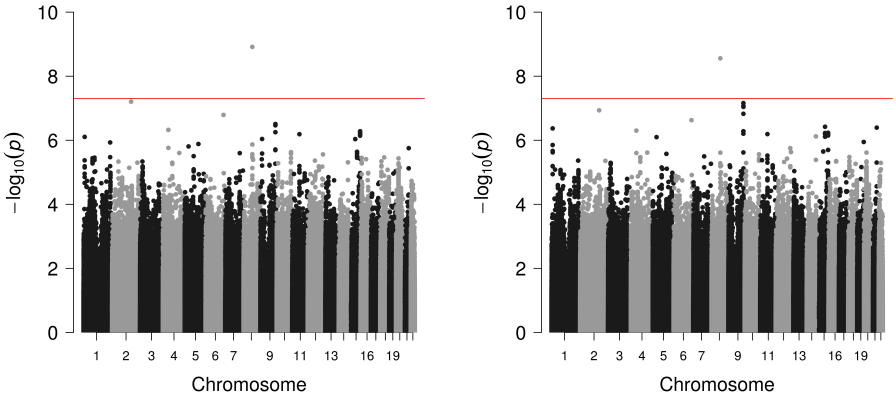
(a) (b)

Figure 6.9: Q-Q plots for longitudinal SNP effect: a) QuickMix, b) CTS. MAF > 0.01.



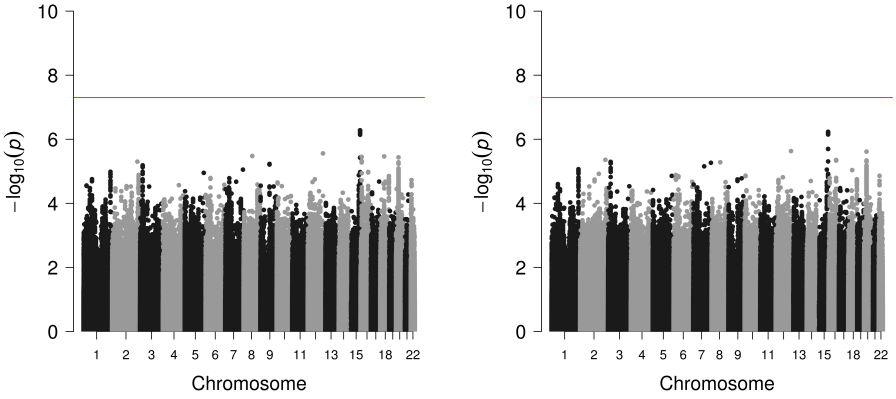
(a) (b)

Figure 6.10: Q-Q plots for longitudinal SNP effect: a) QuickMix, b) CTS. MAF > 0.05.



(a) (b)

Figure 6.11: Manhattan plots: a) QuickMix, b) CTS. MAF > 0.01.



(a) (b)

Figure 6.12: Manhattan plots: a) QuickMix, b) CTS. MAF > 0.05.

No significant SNPs were identified for the cross-sectional SNP effect. The Q-Q plot was slightly inflated, which was probably due to population stratification, which was not taken into account in our model.

Plotting a Q-Q plot takes around 15 minutes and a Manhattan plot around 25 minutes. Additionally, it is memory demanding and cannot be done on a PC without a large RAM. We propose an efficient approach to Manhattan plotting using the smoothing techniques as described in Eilers and Goeman (2004). A "smooth" Manhattan plot can be plotted within a couple of minutes using a lot less memory than the traditional approach. An example of such a plot for 2 Chromosomes is displayed in Figure 6.13.

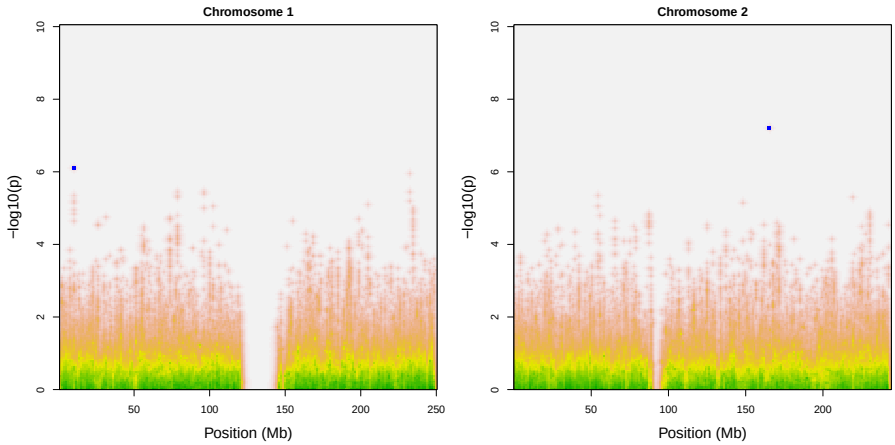


Figure 6.13: Fast approach for Manhattan plot using smoothing techniques for the first 2 Chromosomes.

6.7 Conclusions and Discussion

We have analyzed the longitudinal BMD data from the Rotterdam Study, searching for associations of the evolution of the BMD over time with around 29 million SNPs. Without our methods this task would take 1-2 years on one CPU. We reduced the time to 21 hours for the QuickMix algorithm and 4 hours for the conditional two-step approach.

The precision is higher for the QuickMix algorithm than for the CTS. This is due to the fact that the latter is in fact approximating the conditional linear mixed model. This has been discussed in detail in **Chapter 3** and confirmed here on real data. For very low MAFs, the results from the CLMM were very different than for the LMM (plot not shown). Most of the times the p-values were much larger, suggesting the lack of power of the CLMM to detect the effect of the SNP on the slope when the variability of the SNP is essentially zero. Additionally, the results from the conditional two-step were different from the conditional mixed model. Nevertheless, the SNPs with very low MAFs are filtered out before summarizing the results.

We are still working on additional speed improvements. Other types of array-binary files can be investigated, such as **ff**, **ncdf** or **hdf5** files. Fast SNP data access is vital in achieving the maximum speed of the algorithms.

Acknowledgment

We thank Maarten Kooijman for his technical support while we performed the analyses on the Epib-genstat server.

Chapter 7

Conclusions, Discussion and Further Research

In this thesis, we presented new computational approaches to genome-wide association studies. Our methodology improves computations for cross-sectional continuous and binary outcomes as well as for continuous longitudinal responses. We exploited two approaches to speed up GWA computations. In the first approach, we make use of approximations. Millions of SNPs are investigated in GWAS, but we expect at least 99% of them not to be significant. Approximations facilitate the identification of significant (or close to significant) variants in a rapid manner. Since SNP effects are very small, very precise results could be obtained. In the second approach, we improved directly the speed of the algorithms. Many functions in popular software have been implemented without considering such large-scale analyses and are therefore often not fully optimized. We used the fact that millions of models fitted in GWAS are very similar differing only in the SNP values. This facilitated storing common parts of the solution and hugely increased the speed of the whole analysis.

We also discussed the importance of the organization of the SNP data and proposed efficient solutions. Below we describe the main conclusions of each chapter and directions for future research. We close this chapter with general conclusions on the importance of improving computations in the GWA area.

In **Chapter 2** we proposed the conditional two-step (CTS) approach which approximates the p-value for the longitudinal SNP effect. We were motivated by the bone mineral density data from the Rotterdam Study and HapMap imputed SNPs. A GWA analysis would take at least a month on a single computer. Our research is based on the fact that if a SNP is omitted from the mixed model, its effect is incorporated in the random effects: the cross-sectional effect is seen in random intercepts and the longitudinal effect is seen in random slopes. Next, inspired by the conditional linear mixed model we related unbiased individual trajectories of the outcome to SNPs. In this manner, we reduced the mixed model computations to fitting simple linear regressions, speeding up the computations around 200 times. The CTS proved to be superior to other approximating strategies that were known to us, either from the literature, or through discussions with other scientists. We obtained

essentially exact p-values for balanced data and a very close approximation for unbalanced data.

Satisfactory performance of the CTS approach motivated us to explore further properties of that method. This was the focus of **Chapter 3** in which we described an extensive simulation study. We compared the CTS approach to the two-step method through many data scenarios. In this chapter, we also approached the problem analytically. Due to the complexity of the analytical derivations we limited ourselves to balanced data, which nonetheless gave us a good insight into the properties of the methods and we were able to prove intuitive the relationships that we discovered in **Chapter 2**. We proved the equality of the p-values between the linear mixed model and the conditional two-step for balanced data and derived a relationship between the linear mixed model and the two-step approach. The simulations allowed us to explore this relationship further, as well as assess the performance of the methods for various scenarios of unbalanced data. We showed that the CTS approach remains a solid approximating method unlike the TS which, in certain data situations, leads to unreliable results. The CTS approach may lose up to a few percents of statistical power compared to the linear mixed model, but this happens only in special data situations. We also verified that this power loss is related to the fact that the CTS approach is approximating the conditional linear mixed model, which may give somewhat different results than the linear mixed model.

In **Chapter 4** we switched to cross-sectional outcomes. We speeded up linear regression computations by 60-80 times using semi-parallel regression. The computations were performed exactly and the gain in computation time was achieved by avoiding loops and exploiting smart matrix operations. We used that fact that a SNP is the only part of the regression model that is changing in the equations. By the term "semi-parallel" we emphasize the possibility of quickly solving parallel problems within one CPU. We speeded up fitting one million of regressions from 4 hours to 4 minutes. Semi-parallel regression also brought a great benefit to the conditional two-step approach, where the second step can be now performed extremely fast. The combination of the conditional two-step approach with semi-parallel regression speeded up the computations by a factor of 50000 even. In **Chapter 4**, we also improved the logistic regression computations. Here, we exploited the iteratively reweighted least squares algorithm which is equivalent to maximum likelihood. The combination of the two strategies: approximation of the weights and semi-parallel computations, boosted the speed by a factor of 300 with hardly any loss of precision. In this chapter, we also emphasized the importance of the SNP data organization. We showed that even extremely fast algorithms are useless if genotypes are badly stored. We demonstrated that array-oriented binary files implemented in several R packages serve as a great solution for fast and efficient data access.

In the spirit of fast matrix operations, we moved in **Chapter 5** again to linear mixed models. We presented the QuickMix algorithm which exploits the equivalence of Henderson mixed model equations with the penalized least squares approach. Knowing that the SNP effects are very small we can approximate very precisely the penalty matrix. Next, we performed the calculations avoiding operations on large matrices. The cross-sectional and longitudinal SNP effects can be computed using QuickMix. We speeded up fitting 1 million mixed models from 12 days to half an hour on a single computer. In this chapter we again

stressed the importance of a proper SNP data organization.

The focus of **Chapter 6** was the analysis of longitudinal BMD data and the 1000 Genomes imputed SNPs. The number of investigated SNPs is around 29 million creating a great challenge not only for the computations, but also for managing large outputs and plotting the results. We performed the analysis with the conditional two-step and the QuickMix algorithms and compared the precisions of the results and the time spent on the computations. The conditional two-step approach was around 4 times faster, however, its accuracy with respect to the linear mixed model was somewhat worse than for the QuickMix. As already mentioned, this is due to the fact that the CTS approximates the conditional linear mixed model. Nevertheless, both methods resulted in similar summary plots, such as Manhattan and Q-Q plots. In this chapter, we described benefits of using a database to store the output. We showed that in such large-scale studies the output saved as an ASCII file may be very inefficient in summarizing and evaluating the results. We also demonstrated how to efficiently approach plotting by incorporating smoothing techniques.

A few additional remarks are worth mentioning regarding the use of fast approaches. The conditional linear mixed model may be very useful in situations where values for the baseline covariates are missing for a group of individuals. The conditional model, as independent from the baseline characteristics, incorporates those individuals in the analysis as long as the the outcome and time-varying covariates are available.

On another note, caution should be used in performing meta-analysis merging the results of the analyses from different studies. In case of the conditional two-step approach, estimates and standard errors may be shrunken, which was shown in **Chapter 3**. Thus, performing variance-inverse meta-analysis may lead to spurious results. In **Chapter 3**, we showed how the shrinkage factor can be calculated from the data, allowing transformation of the estimates to the original scale. In case of QuickMix, the approach to meta-analysis remains straightforward.

Many extensions of our research can be considered in the framework of cross-sectional as well as longitudinal outcomes. One important example is the fast analysis of survival outcomes for GWAS analyses. More complex regression models can be considered, such as those including interactions of SNPs with environmental factors. The speedup of more complex mixed models is also a future challenge. Some traits may require going beyond linear relationships with time. Another computational challenge is provided by modeling curvatures through polynomials or splines. Other types of outcomes in the framework of generalized linear mixed models, such as binary outcomes or counts could be considered. Finally, modeling longitudinal data within families demands the incorporation of the between-individual relatedness matrix. This would be one of the most useful extensions of the fast mixed models computations.

In this thesis, we showed how to improve algorithms in order to speed up GWA analysis a few hundreds and sometimes even many thousands times. Our goal was to make the GWA scan feasible on a single computer within a reasonable time. It is important to apply the fastest and the most efficient solutions already at the start of your study, because the amount of data will grow only bigger. This was the case when switching from HapMap to 1000 Genomes imputations. Our experience is, however, that there are as many researchers pursuing computational improvements as there are researchers who believe that

large computing clusters are the solution. We believe that improvements in computational statistics should not be neglected. First of all one should think of financial and even ecological benefits of reducing computational resources. Secondly, the use of clusters has many disadvantages, such as unexpected and sometimes long downtimes due to technical reasons. Moreover, often one has to wait for hours or even days before a job can be submitted. We believe that the methodology presented in this thesis is a useful contribution to the more efficient and elegant computations in the present-day genetic research.

Summary

In the last decade genome-wide association studies (GWAS) have become a popular approach to relate human DNA to traits and diseases. The major tool in GWAS is a single nucleotide polymorphism (SNP), which is defined as a variation in a single nucleotide occurring in at least 1 % of the population. In this hypothesis-free approach, millions of SNPs are tested for their relationship with common diseases. Identified SNPs may pinpoint to new biological pathways. Statistical tools used in genome-wide association testing belong to commonly used methods, such as linear and logistic regression for cross-sectional outcomes, or mixed models for longitudinal outcomes. Nevertheless, the necessity of fitting up to 30 million of such statistical models creates a computational challenge. This thesis focuses on new methodology facilitating speeding up GWA analyses by several orders of magnitude. Our research was motivated by longitudinal bone mineral density data from the Rotterdam Study which serve as a guiding example in this thesis.

Chapter 1 provides a brief introduction to the field of GWAS and a short explanation of the statistical models.

In **Chapter 2**, we present the conditional two-step approach approximating the p-value for the longitudinal SNP effect. We relate slopes of individual trajectories from the reduced model (omitting SNP) with one SNP at a time through simple regression. To prevent possible bias coming from misspecification in the baseline part of the reduced model, we use conditional linear mixed model. Our proposal is compared to a few other procedures through a small simulation study involving balanced and unbalanced data. The performance of our method is also evaluated using the Rotterdam Study data. We show that the p-values given by the conditional two-step approach are very accurate and are obtained even 200 times faster than with standard R functions.

Chapter 3 elaborates further on the conditional two-step approach. We also compare it to the two-step approach based on the linear mixed model. Analytical derivations for the balanced data and an extensive simulation study show the performance of the methods. In particular, we show that our method does not inflate probability of type I error and the loss of power is minimal.

Chapter 4 focuses on continuous and binary cross-sectional outcomes. We propose semi-parallel computations to speed up linear regression. The method is based on the fact that outcome and all additional covariates, except the SNP covariate, are the same in all GWA models. By avoiding loops and using optimized matrix operations the regression estimates and standard errors are obtained 60-80 times faster. The method is adapted

to logistic regression by exploiting that the iteratively reweighted least squares method is equivalent to maximum likelihood. Using the fact that SNP effects are very small we can precisely approximate the weights from the model without a SNP. This way the iterative procedure is reduced to a semi-parallel weighted regression. This approach combined with the conditional two-step approach, allows its second step to be performed extremely fast. In **Chapter 4**, we also elaborate on the importance of the SNP data organization. We show that fast algorithms must be combined with a rapid access to the data. Solutions based on the array-oriented binary files, implemented in a couple of R packages, are discussed.

In **Chapter 5**, the QuickMix algorithm, based on writing the Henderson equations as the penalized least-squares problem, is presented. In this way, the penalty matrix is treated as known and taken from the model without a SNP. This is reasonable since in a typical GWAS genetic effects are very small. In the QuickMix algorithm, operations on large matrices are avoided. We explore the structure of the equations and solve them in a semi-symbolic way. Using QuickMix, we solve cross-sectional and longitudinal SNP effects 1000 faster than standard R functions.

Chapter 6 focuses on the analysis of the longitudinal bone mineral density data from the Rotterdam Study. This involves almost 30 million SNPs coming from 1000 Genomes imputations. Such large-scale data create a challenge not only for the mixed model analysis, but also for other practical aspects. We discuss the issue of managing a large output, that can be very inefficient when saved as an ASCII file. We propose using an SQL database allowing quicker access to the results. Moreover, efficient techniques to plotting Manhattan and Q-Q plots are discussed. Both, the conditional two-step and the QuickMix, are applied. Their speed and performance are thoroughly discussed.

Chapter 7 provides the summary of the findings in each chapter indicating directions for future research.

Samenvatting

In het afgelopen decennium hebben genoombrede associatiestudies (GWAS) sterk aan populariteit gewonnen om het menselijk DNA in verband te brengen met kenmerken en ziektes. Het belangrijkste hulpmiddel in GWAS is een enkelvoudige-nucleotide polymorfie (SNP). Deze wordt omschreven als een variatie in een enkelvoudige-nucleotide die in ten minste 1% van de bevolking voorkomt. In deze benadering, vrij van iedere hypothese, worden miljoenen SNP's getoetst op hun relatie met veel voorkomende ziekten. SNP's die zo geïdentificeerd worden, kunnen duiden op nieuwe biologische paden. De statistische methoden die voor deze associatietoetsen gebruikt worden, zijn klassieke statistische technieken, zoals lineaire en logistische regressie voor cross-sectionele uitkomsten en gemengde modellen voor longitudinale uitkomsten. Het is een uitdaging op computationeel gebied om van 30 miljoen statistische modellen de waarden van de parameters te vinden. Dit proefschrift gaat daarom over nieuwe methoden die de snelheid van GWA analyses drastisch weten te verhogen. Dit onderzoek is gebaseerd op de longitudinale botmineraaldichtheid data zoals gemeten in de Rotterdam Studie.

Hoofdstuk 1 introduceert GWAS en geeft een korte beschrijving van de statistische modellen.

In hoofdstuk 2 presenteren we de voorwaardelijke tweetrapsbenadering die het longitudinale SNP-effect bij benadering berekent. Hiervoor relateren we de hellingen van individuele banen in het model zonder SNP-effecten met de SNP. Deze techniek is gebaseerd op het voorwaardelijke lineaire gemengde model. In een simulatiestudie hebben we de prestaties van deze techniek vergeleken met andere tweetrapstechnieken op zowel gebalanceerde als ongebalanceerde data. De prestaties van onze methode werden ook geëvalueerd door botmineraaldichtheid data van Rotterdam Studie. We tonen aan dat de p-waarden verkregen door onze techniek zeer nauwkeurig zijn en tot 200 keer sneller berekend kunnen worden dan door het gebruik van de standaardfuncties in R.

Hoofdstuk 3 gaat verder in op de voorwaardelijke tweetrapsbenadering. We vergelijken deze met een gelijkwaardige methode gebaseerd op het lineaire gemengde model. Via analytische afleidingen en een uitgebreide simulatiestudie, verkennen we de prestaties van de methode in verschillende datascenario's en tonen we de voordelen van het gebruik van het voorwaardelijke lineaire gemengde model. In het bijzonder laten we zien dat onze methodiek de kans op een type I fout niet verhoogt en dat het verlies aan onderscheidingsvermogen minimaal is.

In hoofdstuk 4 ligt de focus op cross-sectionele studies met continue en binaire uitkomst-

variabelen. We stellen semi-parallelle computaties voor om lineaire regressie te versnellen. Deze methode is gebaseerd op het feit dat de uitkomst en alle covariabelen, behalve de SNP variabele, in alle GWA modellen dezelfde zijn. Door lussen te vermijden en specifieke operaties op matrices toe te passen verkrijgen we de exacte regressiecoëfficiënten en de standaardfouten, 60 tot 80 keer sneller. Een aangepaste versie van deze methode voor logistische regressie verkregen we door gebruik te maken van de iteratief gewogen kleinste kwadraten techniek die gelijkwaardig is aan het bepalen van de meest aannemelijke schatter. Omdat SNP-effecten zeer gering zijn, kunnen we nauwkeurig de wegingsfactoren van het model zonder de SNP variabele benaderen. Op deze manier wordt de iteratieve procedure teruggebracht tot semi-parallelle regressie waarbij de wegingsfactor berekend wordt. Semi-parallelle regressie is ook nuttig voor de voorwaardelijke tweetrapsbenadering, omdat die het mogelijk maakt de tweede trap extreem snel uit te voeren. In Hoofdstuk 4 tonen we ook het belang van een efficiente dataorganisatie voor GWAS aan. We laten zien dat snelle algoritmes gecombineerd moeten worden met een optimale dataopslag. Oplossingen gebaseerd op de array-georiënteerde binaire files die geïmplementeerd zijn in een aantal R pakketen, worden hier ook besproken.

In hoofdstuk 5 presenteren we een andere benadering om de regressiecoëfficiënten van lineaire gemengde modellen snel te bepalen. Het QuickMix algoritme is gebaseerd op interpretatie van de Henderson-vergelijkingen als een 'penalized' kleinste-kwadraten probleem. De 'penalized' matrix, verkregen uit het model zonder SNP, wordt daarna als bekend verondersteld. Dit is een zeer redelijke veronderstelling gezien de voor GWAS karakteristieke geringe genetische effecten. In het QuickMix algoritme vermijden we operaties op grote matrices. We onderzoeken de structuur van de vergelijkingen die we op semi-symbolische wijze oplossen. We illustreren de hoge nauwkeurigheid van onze algoritme die cross-sectionele en longitudinale SNP-effecten 1000 keer sneller berekent dan de standaardfuncties in R.

In hoofdstuk 6 ligt de focus op de analyse van de longitudinale botmineraaldichtheiddata van de Rotterdam Studie en van bijna 30 miljoen SNP's afkomstig uit imputaties van de data van het 1000 Genomes project. Data op zo'n grote schaal is niet alleen een uitdaging voor de analyse met het gemengde model maar ook voor andere, praktische aspecten. We bespreken het probleem van het beheren van een grote output, dat zeer inefficiënt kan zijn wanneer deze als ASCII file opgeslagen wordt. We stellen voor om een SQL-database te gebruiken, die snellere toegang tot de resultaten geeft. Verder worden hier efficiënte technieken besproken voor het maken van Manhattan plots en Q-Q plots. Onze snelle benaderingen, de voorwaardelijke tweetrapsbenadering en de QuickMix, zijn hierop toegepast en hun snelheid en prestaties worden uitgebreid besproken.

In hoofdstuk 7 vatten we onze bevindingen samen en doen we voorstellen voor toekomstig onderzoek.

PhD Portfolio

Conferences

- International Society for Clinical Biostatistics (ISCB), Montpellier, 2010
- International Workshop on Statistical Modelling, Valencia, 2011
- ISCB, Bergen, 2012
- European Mathematical Genetics Meeting (EMGM), Leiden, 2013
- Bayes Conference, Rotterdam, 2013
- EMGM, Cologne, 2014

Conference posters and presentations

- "Computational fast approaches for genomewide association analysis in longitudinal studies" (poster, IWSM 2011)
- "Fast linear mixed model computations for GWAS with longitudinal data using conditional two-step approach" (presentation, ISCB 2012)
- "Fast computations in GWAS - semi parallel linear and logistic regression" (presentation, EMGM 2013)
- "Fast mixed models for GWAS with longitudinal data using QuickMix" (presentation, EMGM 2014)

PhD training

- Frailty models, Nihes, Erasmus Medical Centre, 2009
- Summer Programme, Nihes, Erasmus Medical Centre, 2010
- SNP's and human diseases, MolMed, Erasmus Medical Centre, 2010
- Repeated measures analysis, Nihes, Erasmus Medical Centre, 2010

- Statistical Computing with R, Statistical Science for the Life and Behavioural Sciences, Leiden, 2010
- Longitudinal data analysis, Statistical Science for the Life and Behavioural Sciences, Leiden, 2010
- Bayesian Biostatistics, Statistical Science for the Life and Behavioural Sciences, Leiden, 2010
- The craft of smoothing, Nihes, Erasmus Medical Centre, 2010
- An Introduction to the Joint Modeling of Longitudinal and Survival Outcomes, Erasmus Medical Centre, 2013

Webinar

- Fast Semi-Parallel Linear and Logistic Regression for Genome-Wide Association Studies. Webinar for Extension - Plant Breeding and Genomics, 2013

Teaching

- Classical Methods in Biostatistics (practicals), 2009-2013
- Modern Methods in Biostatistics (practicals), 2009-2013
- Basic Introduction Course to SPSS, 2010-2014
- Repeated measures analysis (practicals), 2011-2012

Acknowledgments

"No one can whistle a symphony. It takes a whole orchestra to play it" (H.E. Luccock). The same can be said about obtaining a PhD degree, which is always the result of very hard work of several people. In my case, the orchestra turned to be relatively large, with three promoters, one co-promoter, one collaborator, and me, the PhD student.

In Poland we have the saying: "Where there are six cooks, there is no meal". Imagine the odds that I would end up in a group of six people working on one PhD project. However, very fortunately, this time the polish saying proved not to be correct. We did cook the meal.

Henry Ford said: "Coming together is a beginning. Keeping together is progress. Working together is success." I realized the full meaning of this when writing this thesis, and throughout my PhD project.

I would like to take the opportunity to thank all the promoters and collaborators: **Emmanuel Lesaffre, André Uitterlinden, Patrick Groenen, Ferndando Rivadeneira, and Paul Eilers** for their efforts in making my PhD possible.

I want to thank especially the two members of our team.

Firstly, I would like to express my gratitude to **Emmanuel Lesaffre** for giving me a chance to join the department of Biostatistics and for supervising my work during these five years. After I came to the Netherlands, I quickly realized how much I had to learn. Dear Emmanuel, your guidance, especially at the beginning of my studies, was crucial in going on the right path with my research. Sometimes a PhD track can be hilly (even in Holland) and we experienced this at times. Nevertheless, I will remain greatly indebted for the opportunity you have given me, for your guidance and your criticism.

Secondly, my special thanks goes out to a person who in Holland would be called a "stille kracht". **Paul Eilers**, initially a professor sharing the office with me and three other PhD students, turned out to be not only a great source of jokes and comments about plots on our screens, but also a great source of knowledge and inspiration. Dear Paul, a big part of this thesis is the result of your enthusiasms and ideas, which you passed on to me during the many talks we had together. You provided me with everything what a PhD student needs: inspiration and confidence, as well as the occasional critical word and the motivation to work hard. Learning from you has been an honor and a pleasure and your attitude of turning research into fun has been very soothing at times. I can see you reading this, thinking, if only my compliments could turn into dropjes and beer.

The Department of Biostatistics is a group of people coming from all over the world, and I found it a friendly and helpful working environment.

It has a special meaning to me to acknowledge my very first colleague here: **Marek Molas** (1979-2013). He was also from Poland and quickly became my good friend. He was incredibly helpful with getting used to the new world of research and with setting up my life in Holland. It was always great fun to share a laugh about the jokes that only polish people can understand. Although he will not see me graduate, his kindness and helpfulness will never be forgotten.

Many thanks also to my great office mates, especially to **Johan** who sat opposite me for four years. Johan, thank you for all your help and all the non-statistical chats through the years. Thanks to Johan, **Siti** and **Li** our office was always "gezellig". My other colleagues: **Sten, Magda, Dimitris, Kazem, Nahid, Susan, Elrozy, and Nicole** were always very helpful and a great companion for a chat about statistics and more than that. Dear Dimitris, you always found time for me when I needed statistical advice. I really appreciate that.

And of course, many thanks to **Eline**, who is always helpful with administrative issues and who always knows which gift is fun. I do hope to get that Bear after the graduation, Eline!

Life is not only work, how fortunately. Time after work would not be so much relaxing for me if not my great friends that I made in Holland. A lot of them were also PhD students in Erasmus MC. **Katharina** and **Ivana**, thanks for all the time we spent together. Special thanks to the two funniest people I have met here. Dear **Rachel** and **Alex**, I would never experience how everything (and anything) can become funny after few beers, if I would not have you around during the last couple of years. You were there for me to talk about serious things and silly stuff. I am glad that our friendship continues even now that our PhD adventures are over.

Even when living far from the home country, it is still invaluable to feel support from the family. I would like to thank my mum and my older brother for raising me in a house full of books and for their encouragement to study hard and to pursue my dreams. Their unconditional faith in me was very motivating. Mamo, Dziękuję Ci za wszystko! Dear Marcin, you were a great example for me of motivated and hard-working person. Although thousands of kilometers are dividing us for most of the year, I am very glad that you are celebrating my graduation with me.

Finally, dear Joris, thank you for your endless support and incredible patience. Thank you for sharing with me "the good, the bad and the ugly" during the last few years. Thank you for your optimism and faith in me. If that is true that laughing prolongs life, I will grow very old. Thank you for your interest in my work, even though it is so far from your world. And most important, thank you for making Holland feel like home to me.

I think that one of the purposes of writing the Acknowledgments is to realize how many great people contribute to our success. I definitely realized that today.

10 August 2014

About the author

Karolina Sikorska was born in 1984 in Bydgoszcz, Poland. After finishing her primary school she joined a special high-school to prepare her for medical studies. After 4 years she decided to study mathematics instead. She joined the Gdańsk University of Technology. After 3 years of general math she chose to specialize in financial mathematics and also followed a few statistical courses. In October 2008 she received her Master's Degree. Her master's thesis was about the analysis of categorical data with examples of medical data. Combining her interests in statistics and medical sciences she pursued a PhD in biostatistics. In April 2009 she joined the Department of Biostatistics in Rotterdam as a PhD student. In the meantime she took many courses in biostatistics. As of June 2014 she is working as a statistician at the Biometrics Department of the Netherlands Cancer Institute in Amsterdam.

List of publications

Papers published in peer-reviewed journals

Sikorska K., Rivadeneira, F., Groenen, P.J.F., Hofman, A., Uitterlinden, A., Eilers, P.H.C., Lesaffre, E. (2013) Fast linear mixed model computations for genome-wide association studies with longitudinal data. *Statistics in Medicine*, 32(1): 165-180

Sikorska K., Lesaffre, E., Groenen, P.J.F., Eilers, P.H.C. (2013) GWAS on your notebook: fast semi-parallel linear and logistic regression for genome-wide association studies. *BMC Bioinformatics*, 14(1):166

Kekelidze, M., Dwarkasing, R.S., Dijkshoorn, M.L., Sikorska, K., Verhagen, P.C., & Krestin, G.P. (2010) Kidney and urinary tract imaging: triple-bolus multidetector CT urography as a one-stop shopprotocol design, opacification, and image quality Analysis 1. *Radiology*, 255(2), 508-516.

Lima, A., van Bommel, J., Sikorska, K., van Genderen, M., Klijn, E., Lesaffre, E., Ince, C. & Bakker, J. (2011) The relation of near-infrared spectroscopy with changes in peripheral circulation in critically ill patients. *Critical Care Medicine*, 39(7), 1649-1654.

Submitted manuscripts

Sikorska K., Lesaffre, E., Groenen, P.J.F., Eilers, P.H.C. More GWAS on your notebook: fast mixed models for longitudinal phenotypes using QuickMix.

Sikorska K., Eilers, P.H.C., Rivadeneira, F., Uitterlinden, A., Lesaffre, E. GWAS with longitudinal phenotypes - performance of approximate procedures.

Conference papers

Sikorska, K., Groenen, P.J.F., Rivadeneira, F., Eilers, P.H.C., Lesaffre, E. (2011) Fast genome-wide association analysis in longitudinal studies. *Proceedings of the 26th International Workshop on Statistical Modelling*

Bibliography

1000 Genomes Project Consortium and others (2012). An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491(7422):56–65.

Adler, D., Glser, C., Nenadic, O., Oehlschlgel, J., and Zucchini, W. (2012). *ff: memory-efficient storage of large data on disk and fast access functions*. R package version 2.2-7.

Agresti, A. (2002). *Categorical Data Analysis*. WILEY SERIES IN PROBABILITY AND STATISTICS.

Aloia, J. F., Vaswani, A., Ross, P., and Cohn, S. H. (1990). Aging bone loss from the femur, spine, radius, and total skeleton. *Metabolism*, 39(11):1144 – 1150.

Amin, N., Byrne, E., Johnson, J., Chenevix-Trench, G., Walter, S., Nolte, I., Vink, J., Rawal, R., Mangino, M., Teumer, A., et al. (2012). Genome-wide association analysis of coffee drinking suggests association with CYP1A1/CYP1A2 and NRCAM. *Molecular Psychiatry*, 17(11):1116–1129.

Aulchenko, Y., Ripke, S., Isaacs, A., and Van Duijn, C. (2007). GenABEL: an R library for genome-wide association analysis. *Bioinformatics*, 23(10):1294–1296.

Aulchenko, Y., Struchalin, M., and van Duijn, C. (2010). ProbABEL package for genome-wide association analysis of imputed data. *BMC Bioinformatics*, 11(1):134.

Bates, D., Maechler, M., Bolker, B., and Walker, S. (2014). *lme4: Linear mixed-effects models using Eigen and S4*. R package version 1.1-7.

Clayton, D. (2012). *snpStats: SnpMatrix and XSnpMatrix classes and methods*. R package version 1.6.0.

Diggle, P., Heagerty, P., Liang, K., and Zeger, S. (2002). Analysis of longitudinal data. Oxford University Press. *New York*, pages 141–168.

Eilers, P. H. and Goeman, J. J. (2004). Enhancing scatterplots with smoothed densities. *Bioinformatics*, 20(5):623–628.

Estrada, K., Abuseiris, A., Grosveld, F., Uitterlinden, A., Knoch, T., and Rivadeneira, F. (2009). GRIMP: a web-and grid-based tool for high-speed analysis of large-scale genome-wide association using imputed data. *Bioinformatics*, 25(20):2750–2752.

Feero, W. G., Guttmacher, A. E., and Manolio, T. A. (2010). Genomewide association studies and assessment of the risk of disease. *New England Journal of Medicine*, 363(2):166–176.

Frazer, K., Ballinger, D., Cox, D., Hinds, D., Stuve, L., Gibbs, R., Belmont, J., Boudreau, A., Hardenbol, P., Leal, S., et al. (2007). A second generation human haplotype map of over 3.1 million SNPs. *Nature*, 449(7164):851–861.

Gibbs, R. A., Belmont, J. W., Hardenbol, P., Willis, T. D., Yu, F., Yang, H., Ch'ang, L.-Y., Huang, W., Liu, B., Shen, Y., et al. (2003). The international HapMap project. *Nature*, 426(6968):789–796.

Gurrin, L. C., Scurrah, K. J., and Hazelton, M. L. (2005). Tutorial in biostatistics: spline smoothing with linear mixed models. *Statistics in Medicine*, 24(21):3361–3381.

Hannan, M. T., Felson, D. T., Dawson-Hughes, B., Tucker, K. L., Cupples, L. A., Wilson, P. W. F., and Kiel, D. P. (2000). Risk factors for longitudinal bone loss in elderly men and women: The Framingham Osteoporosis Study. *Journal of Bone and Mineral Research*, 15(4):710–720.

Helms, R. (1992). Intentionally incomplete longitudinal designs: I. methodology and comparison of some full span designs. *Statistics in Medicine*, 11(14-15):1889–1913.

Henderson, C. R. (1975). Best linear unbiased estimation and prediction under a selection model. *Biometrics*, pages 423–447.

Henderson, C. R., Kempthorne, O., Searle, S. R., and Von Krosigk, C. (1959). The estimation of environmental and genetic trends from records subject to culling. *Biometrics*, 15(2):192–218.

Hindorff, L., Junkins, H., Mehta, J., Manolio, T., et al. (2010). A catalog of published genome-wide association studies. *National Human Genome Research Institute (Available at http://www. genome. gov/gwastudies)*.

Hofman, A., Murad, S. D., Van Duijn, C. M., Franco, O. H., Goedegebure, A., Ikram, M. A., Klaver, C. C., Nijsten, T. E., Peeters, R. P., Stricker, B. H. C., et al. (2013). The Rotterdam Study: 2014 objectives and design update. *European Journal of Epidemiology*, 28(11):889–926.

Jacqmin-Gadda, H., Sibillot, S., Proust, C., Molina, J.-M., and Thiébaut, R. (2007). Robustness of the linear mixed model to misspecified error distribution. *Computational Statistics & Data Analysis*, 51(10):5142–5154.

Jones, G., Nguyen, T., Sambrook, P., Kelly, P. J., and Eisman, J. A. (1994). Progressive loss of bone in the femoral neck in elderly people: longitudinal findings from the Dubbo osteoporosis epidemiology study. *BMJ*, 309(6956):691–695.

Kenward, M. G. and Roger, J. H. (1997). Small sample inference for fixed effects from restricted maximum likelihood. *Biometrics*, pages 983–997.

Kerner, B., North, K., and Fallin, M. (2009). Use of longitudinal data in genetic studies in the genome-wide association studies era: Summary of Group 14. *Genetic Epidemiology*, 33(S1):S93–S98.

Laird, N. M. and Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics*, pages 963–974.

Li, Y., Willer, C., Ding, J., Scheet, P., and Abecasis, G. (2010). MaCH: using sequence and genotype data to estimate haplotypes and unobserved genotypes. *Genetic Epidemiology*, 34(8):816–834.

Li, Y., Willer, C., Sanna, S., and Abecasis, G. (2009). Genotype imputation. *Annual Review of Genomics and Human Genetics*, 10:387.

Liang, K. and Zeger, S. (1986). Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1):13–22.

Lippert, C., Listgarten, J., Liu, Y., Kadie, C., Davidson, R., and Heckerman, D. (2011). Fast linear mixed models for genome-wide association studies. *Nature Methods*, 8(10):833–835.

Littell, R. (2006). *SAS for Mixed Models*. SAS institute.

Little, R. and Rubin, D. (1987). *Statistical analysis with missing data*, volume 4. Wiley New York.

Little, R. J. and Rubin, D. B. (1989). The analysis of social science data with missing values. *Sociological Methods & Research*, 18(2-3):292–326.

Mitchell, B. D. and Yerges-Armstrong, L. M. (2011). The genetics of bone loss: Challenges and prospects. *The Journal of Clinical Endocrinology and Metabolism*, 96(5):1258–1268. PMID: 21346070.

Morley, A. P., Narayanan, M., Mines, R., Molokhia, A., Baxter, S., Craig, G., Lewis, C. M., and Craig, I. (2012). AVPR1A and SLC6A4 polymorphisms in choral singers and non-musicians: A gene association study. *PloS One*, 7(2):e31763.

Orelien, J. G. and Edwards, L. J. (2008). Fixed-effect variable selection in linear mixed models using R2 statistics. *Computational Statistics & Data Analysis*, 52(4):1896–1907.

Pearson, T. and Manolio, T. (2008). How to interpret a genome-wide association study. *JAMA: The Journal of the American Medical Association*, 299(11):1335–1344.

Pe'er, I., Yelensky, R., Altshuler, D., and Daly, M. (2008). Estimation of the multiple testing burden for genomewide association studies of nearly all common variants. *Genetic Epidemiology*, 32(4):381–385.

Pierce, D. (2011). *ncdf: Interface to Unidata netCDF data files*. R package version 1.6.6.

Pinheiro, J., Bates, D., DebRoy, S., Sarkar, D., and R Core Team (2013). *nlme: Linear and Nonlinear Mixed Effects Models*. R package version 3.1-111.

Purcell, S., Neale, B., Todd-Brown, K., Thomas, L., Ferreira, M., Bender, D., Maller, J., Sklar, P., De Bakker, P., Daly, M., et al. (2007). Plink: a tool set for whole-genome association and population-based linkage analyses. *The American Journal of Human Genetics*, 81(3):559–575.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Riggs, B., Wahner, H., Seeman, E., Offord, K., Dunn, W., Mazess, R., Johnson, K., and Melton III, L. (1982). Changes in bone mineral density of the proximal femur and spine with aging: differences between the postmenopausal and senile osteoporosis syndromes. *Journal of Clinical Investigation*, 70(4):716.

Rivadeneira, F., Styrkársdottir, U., Estrada, K., Halldórsson, B., Hsu, Y., Richards, J., Zillikens, M., Kavvoura, F., Amin, N., Aulchenko, Y., et al. (2009). Twenty bone-mineral-density loci identified by large-scale meta-analysis of genome-wide association studies. *Nature Genetics*, 41(11):1199–1206.

Robins, J., Rotnitzky, A., and Zhao, L. (1994). Estimation of regression coefficients when some regressors are not always observed. *Journal of the American Statistical Association*, pages 846–866.

Robinson, G. (1991). That BLUP is a good thing: The estimation of random effects. *Statistical Science*, 6(1):15–32.

Ruppert, D., Wand, M. P., and Carroll, R. J. (2003). *Semiparametric regression*, volume 12. Cambridge University Press.

Satterthwaite, F. E. (1941). Synthesis of variance. *Psychometrika*, 6(5):309–316.

Sellke, T., Bayarri, M., and Berger, J. (2001). Calibration of ρ values for testing precise null hypotheses. *The American Statistician*, 55(1):62–71.

Shabalin, A. (2012). Matrix eqtl: Ultra fast eqtl analysis via large matrix operations. *Bioinformatics*, 28(10):1353–1358.

Sikorska, K., Lesaffre, E., Groenen, P., and Eilers, P. (2013a). GWAS on your notebook: fast semi-parallel linear and logistic regression for genome-wide association studies. *BMC Bioinformatics*, 14(1):166.

Sikorska, K., Mostafavi, N., Uitterlinden, A. G., Rivadeneira, F., Eilers, P. H., and Lesaffre, E. (2014). GWAS with longitudinal phenotypes - performance of approximate procedures. *European Journal of Human Genetics. Under revision.*

Sikorska, K., Rivadeneira, F., Groenen, P. J., Hofman, A., Uitterlinden, A. G., Eilers, P. H., and Lesaffre, E. (2013b). Fast linear mixed model computations for genome-wide association studies with longitudinal data. *Statistics in Medicine*, 32(1):165–180.

Smith, E., Chen, W., Kähönen, M., Kettunen, J., Lehtimäki, T., Peltonen, L., Raitakari, O., Salem, R., Schork, N., Shaw, M., et al. (2010). Longitudinal genome-wide association of cardiovascular disease risk factors in the Bogalusa heart study. *PLoS Genetics*, 6(9):e1001094.

Van der Loos, M. J., Koellinger, P. D., Groenen, P. J., and Thurik, A. R. (2010). Genome-wide association studies and the genetics of entrepreneurship. *European Journal of Epidemiology*, 25(1):1–3.

Verbeke, G. and Fieuws, S. (2007). The effect of miss-specified baseline characteristics on inference for longitudinal trends in linear mixed models. *Biostatistics*, 8(4):772–783.

Verbeke, G. and Molenberghs, G. (2009). *Linear mixed models for longitudinal data*. Springer.

Verbeke, G., Spiessens, B., and Lesaffre, E. (2001). Conditional linear mixed models. *The American Statistician*, 55(1):25–34.

Vonesh, E. F. and Chinchilli, V. M. (1997). *Linear and nonlinear models for the analysis of repeated measurements*, volume 1. CRC press.

Vonesh, E. F., Chinchilli, V. M., and Pu, K. (1996). Goodness-of-fit in generalized nonlinear mixed-effects models. *Biometrics*, pages 572–587.

Wand, M. P. (2003). Smoothing and mixed models. *Computational Statistics*, 18(2):223–249.

Xu, R. (2003). Measuring explained variation in linear mixed effects models. *Statistics in Medicine*, 22(22):3527–3541.

Zhou, X. and Stephens, M. (2012). Genome-wide efficient mixed-model analysis for association studies. *Nature Genetics*, 44(7):821–824.