Flexible Models for Simple Longitudinal Data

Helen Ogden

University of Southampton, UK

Abstract

We propose a new method for estimating subject-specific mean functions from longitudinal data. We aim to do this in a flexible manner (without restrictive assumptions about the shape of the subject-specific mean functions), while exploiting similarities in the mean functions between different subjects. Functional principal components analysis fulfils both requirements, and methods for functional principal components analysis have been developed for longitudinal data. However, we find that these existing methods sometimes give fitted mean functions which are more complex than needed to provide a good fit to the data. We develop a new penalised likelihood approach to flexibly model longitudinal data, with a penalty term to control the balance between fit to the data and smoothness of the subject-specific mean curves. We run simulation studies to demonstrate that the new method substantially improves the quality of inference relative to existing methods across a range of examples, and apply the method to data on changes in body composition in adolescent girls.

Keywords: Clustered data; Functional principal components analysis; Mixed-effects models; Penalised likelihood; Smoothing; Subject-specific inference

1 Introduction

We consider a simple longitudinal data setup, in which observations are made on each of d subjects, with observations on subject i at n_i time points. We are interested in modelling how the mean response for each subject varies over time.

Writing Y_{ij} for the observation on subject i at time point t_{ij}, we might model

$$Y_{ij} = \mu_i(t_{ij}) + \epsilon_{ij}, \quad i = 1, \ldots, d, \quad j = 1, \ldots n_i, \tag{1}$$

where $\mu_i(.)$ is a function describing how the mean response for subject i varies over time, and $\epsilon_{ij} \sim N(0, \sigma^2)$ are independent error terms. Our interest is in estimating the subject-specific mean functions $\mu_i(.)$. In doing this, we aim to allow flexible dependence on time, and to make use of similarities between these curves to improve estimation.

A large range of methods exist for this problem, and we review some of them in Section 3. The most commonly-used methods are simple mixed-effects models which assume each $\mu_i(.)$ varies linearly with time. In Section 2.1, we motivate the need for more flexible models for longitudinal data, giving a real-data example in which the assumption of linear dependence on time is too restrictive.

To allow flexible dependence on time while still exploiting similarities between the subjects, we focus on methods from functional principal components analysis, in which each subject-specific mean function is expressed as the population mean plus a subject-specific combination of a small number of unknown functions. These functions, the principal component functions, are chosen to be able to best explain the variation between the subject-specific mean functions.

Methods for functional principal components analysis with longitudinal data (often known as sparse functional principal components analysis) have been developed previously (Yao et al., 2005; Di et al., 2009). However, these methods sometimes give fitted mean functions which are more complex than necessary. In Section 2.2, we give an example with data simulated from a simple

random intercept model, with a linear dependence on time, and show that existing methods for sparse functional principal components analysis give wiggly curves for the fitted mean functions. This is problematic because it may lead to the conclusion that there is a complex relationship with time even in cases where the true relationship is linear.

In this paper, we develop a new penalised likelihood approach to flexibly model longitudinal data, with a penalty term to control the balance between fit to the data and smoothness of the subject-specific mean curves. In Section 5, we run simulation studies across a range of examples to demonstrate that the new method substantially improves the quality of inference relative to existing methods, in terms of estimation error (often several times smaller than existing methods) and confidence intervals (with close to correct coverage, in cases where existing methods substantially under-cover).

The setup (1) is a simple one, with normal errors and a single covariate (time). We focus on this simple setup to make the methods development and coding more straightforward, and to make the main ideas easier to understand. The same general ideas have the scope to be applied to more complex modelling problems, and we discuss some possible extensions in Section 7.

2 Motivating examples

2.1 Percent body fat in adolescent girls

A common modelling assumption is that the subject-specific mean functions are linear in time. We first give an example where this assumption is not reasonable, to motivate the development of more flexible models.

We consider data on body fat measurements taken from the MIT Growth and Development Study, originally studied by Phillips et al. (2003). The data is used as an example in Fitzmaurice et al. (2011), and is made available as the `fat` data in the associated ALA R package (Luque and Bates, 2012).

The data contains measurements of percent body fat for 162 girls, in the years before and after menarche (time of first period). Measurements were taken roughly annually, with the last appointment scheduled four years after menarche. Time is rescaled to be relative to time of menarche, so that zero represents reported time of menarche. The number of observations on each subject ranges from 3 and 10, with a mean of 6.5. The data for the first 20 girls is shown in Figure 1. The estimated $\hat{\mu}_i(.)$ from a linear mixed model with random intercepts and slopes is overlaid.

It seems that percentage fat tends to increase over time, but this increase does not appear to be linear. Fitzmaurice et al. (2011) consider a smoothed curve for the mean response over time, similar to Figure 2. On average, percent body fat seems to increase more slowly before menarche than after.

A key advantage of longitudinal data relative to cross-sectional data is that it enables the study of individual trajectories, as well as the average behaviour in the population. There are many possibilities for those individual trajectories consistent with any given population-averaged trajectory (such as the curve in Figure 2). For instance, individuals might all have trajectories of the same shape, shifted up or down by an intercept, or the shape of those trajectories might differ substantially between individuals. In the latter case, we might want describe the similarities and differences between the shapes of individual curves. For instance, we might ask whether there are times at which the percent body fat is increasing rapidly for a large majority of girls, or other times with high variability between girls, with percent body fat increasing rapidly for some and decreasing rapidly for others.

To answer this sort of question, we need to model the individual curves. If there were a large number of observations on each subject, it might be reasonable to simply fit a separate smooth curve for each subject. However, in this case we have a small number of observations per subject, so it is important to share information between subjects to obtain better estimates of the individual curves. For instance, by borrowing information from others we may hope to get a reasonable estimate for the trajectories of subjects 8 and 13, even though we only have a small number of observations on them.
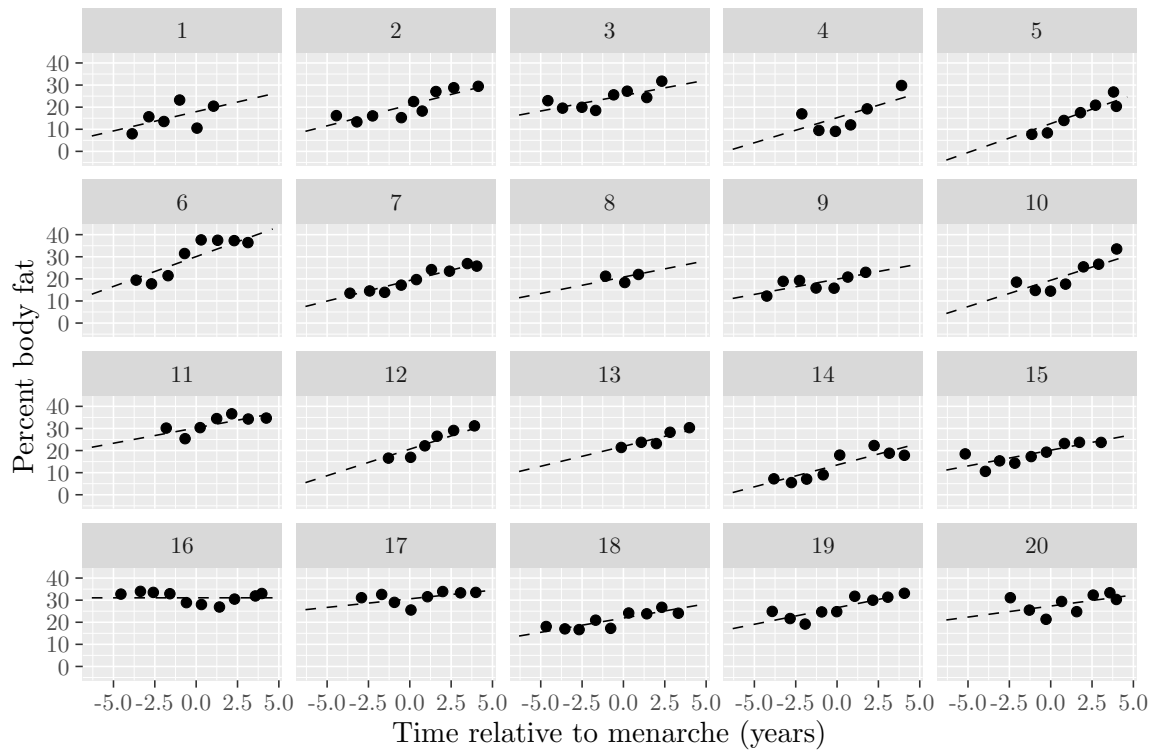
Figure 1: Percentage body fat against time, for the first twenty girls in the `fat` data. The fit from a random slopes model is shown with dashed lines.
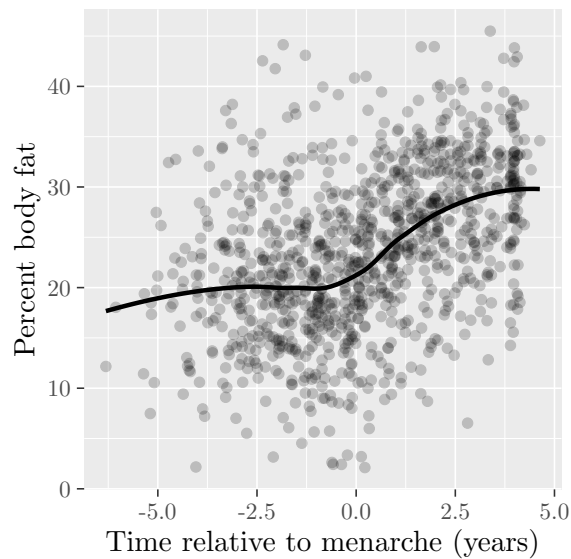

Figure 2: Percentage body fat against time for all girls in the `fat` data, with loess smoothed curve overlaid.
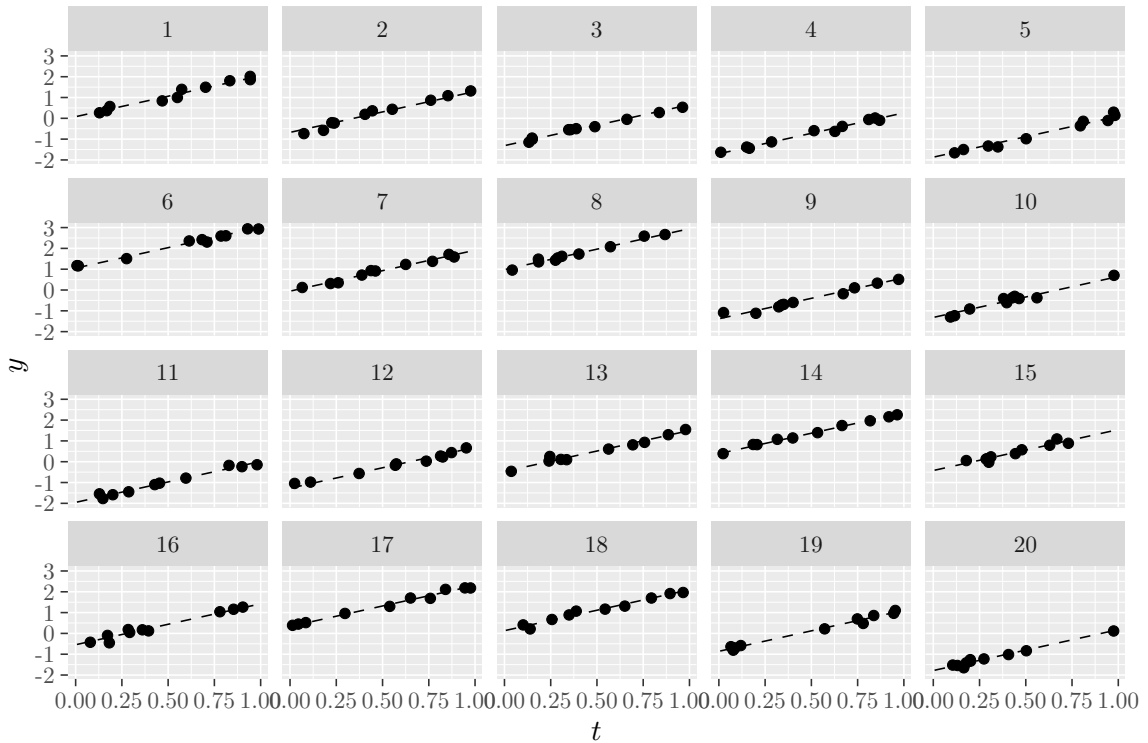
Figure 3: Example simple longitudinal data, with fit from a random intercept model shown with dashed lines.

Linear mixed models (such as the random slopes model shown in Figure 1) allow such information sharing when the individual mean curves $\hat{\mu}_i(.)$ are modelled as straight lines. The aim of this paper is to develop new methods to enable flexible non-linear dependence on time, while sharing information between subjects. We analyse the `fat` data using our new methods in Section 6.

2.2 Simulated data from a random intercept model

To motivate why improvements to current methods for functional principal component analysis are needed, consider the data shown in Figure 3. The fit from a linear random intercept model is show with dashed lines. This simple random intercept model appears to fit the data very well (and in fact the data are simulated from this model).

Figure 4 shows the estimated curve $\hat{\mu}_1(.)$ for the first subject, using three approaches to functional principal component analysis: PACE (Yao et al., 2005) and Di (Di et al., 2009), and the new penalised likelihood method proposed in this paper. We will review the existing approaches in Section 3. To use the PACE and Di methods, we must make some choices about which methods to use for tuning parameter selection. The details of the precise methods used here are described as PACE-BIC and Di-95 in Section 5.1.2.

PACE and Di both estimate $\hat{\mu}_1(.)$ as a wiggly curve, while the estimate from the new penalised likelihood method is very close to the straight line fit from the random intercept model. Estimates of the subject-specific mean curves for the other subjects behave similarly. At first glance, it appears that PACE and Di may be over-fitting to the data: estimating wiggly curves rather than straight lines in order to match the data more closely. However, this is not the case: the residual sum of squares is smaller for the random intercept fit (1.8) than for the PACE (3.0) or Di (3.6) fits.

A desirable property of flexible regression methods is that if a simple linear model fits the
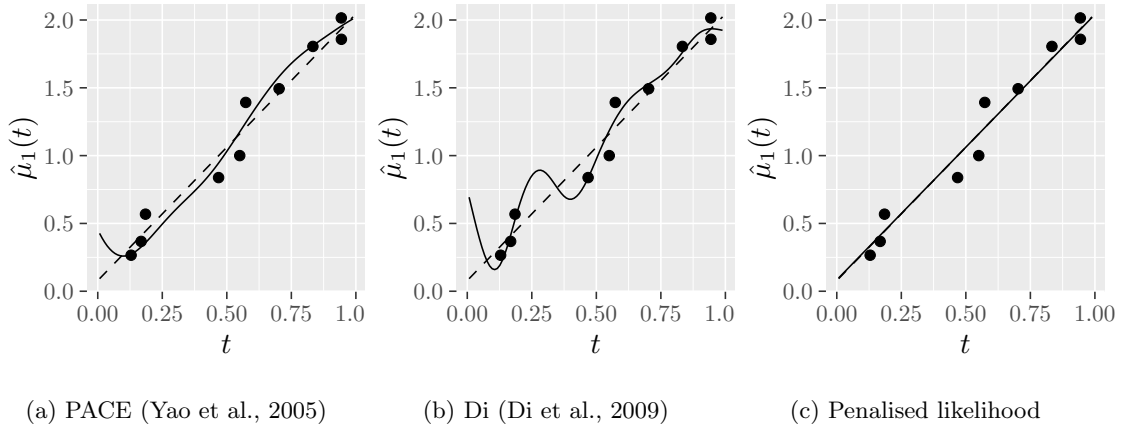
4

(a) PACE (Yao et al., 2005)   (b) Di (Di et al., 2009)   (c) Penalised likelihood

Figure 4: The estimated curve $\hat{\mu}_1(.)$ for the the first subject, for the data shown in Figure 3, using a variety of flexible regression methods (solid lines). The fit from a random intercept model is overlaid with dashed lines. The points show the observations made on the first subject.

data well, the flexible regression model should return a similar fit, close to a straight line. While existing methods based on functional principal components will give consistent estimates of the individual curves given sufficient data, they do not have this property of parsimony. The new penalised likelihood methodology developed here directly penalises average wiggliness of the mean curves, with a smoothing parameter trading off fit to the data against this average wiggliness.

3 Existing modelling approaches

3.1 Flexible models without the longitudinal component

Without the longitudinal component, we could use a penalised spline approach to flexibly model the mean function $\mu(.)$ by

$$\mu(t) = \sum_{j=1}^{n_B} \beta_j b_j(t),$$

where where $b_l(.)$ are spline basis functions. To avoid over-fitting, we could estimate the parameters by using penalised likelihood

$$\ell_p(\beta, \sigma^2) = \ell(\beta, \sigma^2) - \gamma w(\mu), \tag{2}$$

where

$$w(f) = \int_{-\infty}^{\infty} f''(t) dt$$

is wiggliness of a function f and γ is a smoothing parameter, which may be chosen by generalised cross validation or by maximising a marginal likelihood (Wood, 2017). This is not directly useful here, because we wish to model the subject-specific mean, but our proposed approach described in Section 4 involves a penalised likelihood with similar structure to (2).

3.2 Linear mixed models

Linear mixed models provide a relatively simple approach for modelling the subject-specific mean functions $\mu_i(.)$. We could assume a random intercept model

$$\mu_i(t) = \beta_0 + \beta_1 t + u_{0i}, \quad u_{0i} \sim N(0, \sigma_u^2) \tag{3}$$

or a random slopes model

$$\mu_i(t) = \beta_0 + \beta_1 t + u_{0i} + u_{1i}, \quad u_i = (u_{0i}, u_{1i})^T \sim N_2(0, \Sigma_u). \tag{4}$$

5

The random intercept model (3) assumes that each subject deviates from the population-averaged response only by changing the intercept, while the random slopes model (4) allows the mean response curve for each subject to be any straight line.

The simple linear mixed models (3) and (4) make a strong assumption of linearity of the mean curves over time, which is often not appropriate (as for instance in the `fat` data from Section 2.1).

3.3 Linear mixed models with spline basis

More flexibility can be included within the framework of linear mixed models by using a spline basis. We could use a random intercept model

$$\mu_i(t) = \sum_{j=1}^{n_B} \beta_j b_j(t) + u_{0i}, \quad u_{0i} \sim N(0, \sigma_u^2) \tag{5}$$

or the analogue of the random slopes model, which includes a random subject-specific term for each basis coefficient

$$\mu_i(t) = \sum_{j=1}^{n_B} \beta_j b_j(t) + \sum_{j=1}^{n_B} u_{ji} b_j(t) \quad u_i = (u_{1i}, \ldots, u_{n_B i})^T \sim N_{n_B}(0, \Sigma_u). \tag{6}$$

The random intercept model (5) assumes that each subject deviates from the population-averaged curve only by changing the intercept, which may be too simplistic. Model (6) is very flexible (provided that a sufficient number of basis functions are used), but requires large number of observations on each subject in order to estimate the $O(n_B{}^2)$ unknown parameters. We may need to take n_B fairly large to provide a flexible basis, but a large n_B makes the estimation problem worse.

The model (6) does not penalise the wiggliness of the subject-specific mean curves. There are several related approaches which model each subject-specific mean curve as a combination of spline basis functions, including a penalty on the wiggliness of these curves. Pedersen et al. (2019) provides a review, including code to fit these models in the `mgcv` R package (Wood, 2017). Scheipl et al. (2015) describes a spline-based approach, including extensions to more complex problems, available in the `pffr` function of the `refund` R package (Goldsmith et al., 2023). Unlike model (6), the random effects for each spline coefficient are assumed independent in these models, with simple structure $u_i \sim N_{n_B}(0, \sigma_u^2 I)$. This substantially reduces the number of parameters which need to be estimated, but does not allow modelling of dependence between the random coefficients.

3.4 Functional principal components analysis

3.4.1 Motivation

In many cases, there may be variation between the mean curves which is not explained by a change in intercept, but which could be explained in some relatively simple way. For instance, in Figure 5a, the mean curves are clearly related to one another, but an intercept is not enough to explain the variation. In this particular case, the curves are generated according to

$$\mu_i(t) = f_0(t) + u_i f_1(t), \quad u_i \sim N(0, 1), \tag{7}$$

with the function $f_0(.)$ as in Figure 5b and $f_1(.)$ in 5c.

In general, this motivates us to consider

$$\mu_i(t) = f_0(t) + \sum_{k=1}^{K} u_{ki} f_k(t), \quad u_i = (u_{1i}, \ldots, u_{Ki})^T \sim N_K(0, \Sigma_u). \tag{8}$$

where $f_j(.)$ are K unknown functions (which could be written in terms of a spline basis), describing the variation in the subject-specific mean functions. K is the unknown number of functions needed to describe the variation, where in many cases $K \ll n_B$. For instance, a random intercept model

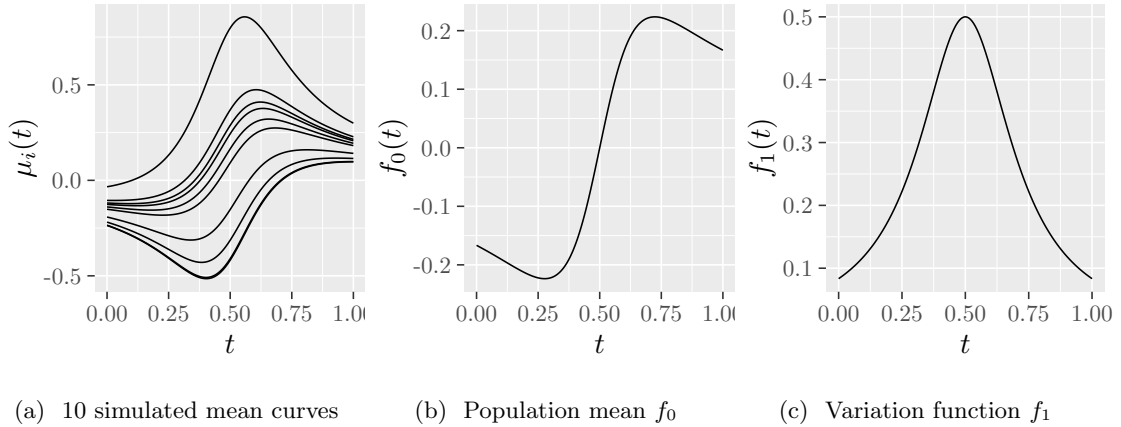| (a) 10 simulated mean curves | (b) Population mean f_0 | (c) Variation function f_1 |

Figure 5: An example of model (7) with one-dimensional variation.

has $K = 1$ variation function (a constant), and a random slopes model has $K = 2$ variation functions (a constant and a linear function).

In the form (8) this model is unidentifiable. To see this, write $f(t) = (f_1(t), \ldots, f_K(t))^T$, so that $\mu_i(t) = f_0(t) + u_i^T f(t)$, and choose any orthogonal matrix Q. Then let

$$u_i^* = Q u_i, \quad f^*(t) = Q f(t),$$

where

$$u_i^* \sim N_K(0, \Sigma_u^*), \quad \Sigma_u^* = Q \Sigma_u Q^T.$$

Since

$$(u_i^*)^T f^*(t) = u_i^T Q^T Q f(t) = u_i^T f(t),$$

$(f_0(t), f(t), \Sigma_u)$ and $(f_0(t), f^*(t), \Sigma_u^*)$ parameterise the same process. We need to impose further constraints on the variation functions and Σ_u to make (8) an identifiable model.

3.4.2 The Karhunen-Loéve decomposition

These ideas are strongly linked to functional principal components analysis (FPCA), which relies on the Karhunen-Loéve decomposition

$$\mu_i(t) = f_0(t) + \sum_{k=1}^{\infty} u_{ki} f_k(t), \tag{9}$$

where u_{ki} are uncorrelated random variables, with $E(u_{ki}) = 0$ and $\mathrm{var}(u_{ki}) = 1$, ordered such that $\lambda_1 \geq \lambda_2 \geq \ldots$. Here $f_1, f_2, \ldots$ are orthonormal functions, that is

$$\langle f_i, f_j \rangle = 0 \text{ for all } i \neq j \in \{1, 2, \ldots, \}$$

and

$$\langle f_k, f_k \rangle = 1 \text{ for all } k \in \{1, 2, \ldots, \}$$

where

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(t) g(t) dt.$$

The functions f_k, $k = 1, \ldots$ are called the functional principal components, or eigenfunctions, of the process.

7

3.4.3 Estimating the functional principal components

The first methods introduced to estimate functional principal components were developed for the case when a large number of observations are made on each subject, at the same dense grid of time points for each subject, as is typically the case for functional data. In longitudinal data, the observation times may be irregular and different for each subject, and the number of observations per subject is often relatively small.

The link between functional data analysis and longitudinal data analysis was described by Yao et al. (2005), who developed the PACE (principal components analysis through conditional expectation) method for estimating the functional principal components for longitudinal data. PACE is a two-step approach: first the mean f_0 and functional principal components $f_1, \ldots, f_k$ are estimated, then the rest of the model is fitted as if these estimated functions were fixed. The global mean f_0 is estimated by pooling across the subjects and fitting a smooth curve. The functional principal components are estimated from a smoothed version of the estimated covariance function. There are various approaches to do this smoothing: PACE uses kernel smoothing, while Di et al. (2009) use penalised splines to smooth the covariance. PACE is available through the `fdapace` R package (Zhou et al., 2022), while the Di et al. (2009) approach is available through the `fpca.sc` function the `refund` R package (Goldsmith et al., 2023). These methods involve two smoothing parameters: one for smoothing the mean, and another for smoothing the covariance. The number of functional principal components K must also be chosen, e.g. to explain a certain proportion of the variation, or by other criteria such as AIC or BIC.

As well as estimating the subject-specific mean functions $\mu_i(.)$, we would like to be able to express our uncertainty in these estimates. Goldsmith et al. (2013) point out that this is problematic for two-step approaches, because these methods do not take into account uncertainty in the estimated functional principal components. Since we are using a full likelihood framework, we may use standard likelihood-based inference methods to quantify uncertainty.

We develop an new penalised likelihood approach to estimating the functional principal components, jointly estimating the mean and functional principal components together with the other parameters of the model. We demonstrate through a range of examples that our penalised likelihood approach provides substantially more accurate estimates of the subject-specific mean curves than two-step approaches, and provides confidence intervals with close to nominal coverage.

4 A penalised likelihood approach

4.1 The model

For some K and functions $f_0, f_1, \ldots, f_K$, we use model (1) with the subject-specific mean functions assumed to be of the form

$$\mu_i(t) = f_0(t) + \sum_{k=1}^{K} u_{ik} f_k(t), \tag{10}$$

where u_{ik} are independent and identically distributed $N(0, 1)$ random variables. Here $f_0, f_1, \ldots, f_k$ and σ^2 are all unknown quantities which must be estimated. For now, we view K as fixed, but will describe methods to choose the value of K in Section 4.7.

To make the model identifiable, we must impose some constraints on $f_1, \ldots f_K$. Motivated by the Karhunen-Loéve decomposition (9), we enforce the constraint that $f_1, \ldots, f_K$ must be orthogonal functions, that is

$$\langle f_i, f_j \rangle = 0 \text{ for all } i \neq j \in \{1, \ldots, K\}. \tag{11}$$

We enforce orthogonality rather than orthonormality because we assume that random effects u_{ik} have unit variance. This is equivalent to modelling $u_{ik} \sim N(0, \lambda_k)$, and enforcing orthonormality. The eigenvalues associated with our model can be found as $\lambda_k = \langle f_k, f_k \rangle = \|f_k\|^2$.

With the orthogonality constraint, the model is nearly identifiable, up to two simple types of transformations of the functional principal components. We can permute the order of $f_1, \ldots, f_K$, or for any k change f_k for $-f_k$ and each u_{ik} for $-u_{ik}$, without altering the overall process. In

theory, we could add extra constraints to make the model fully identifiable: ordering $f_1, \ldots, f_K$ by size, so that

$$\|f_1\| \geq \|f_2\| \geq \ldots \geq \|f_K\|, \tag{12}$$

and deciding on the sign of f_k by insisting that

$$\int f_k(t)dt \geq 0. \tag{13}$$

In practice, we do not enforce these additional constraints, although our optimisation method nearly always gives estimated f_k in order of size, meeting (12). If necessary, it is straightforward to transform and solutions to meet constraints (12) and (13), by permutation and changes of sign.

For $k = 0, \ldots, K$, write $f_{ki} = (f_k(t_{i1}), \ldots, f_k(t_{in_i}))^T$ for the vector which evaluates the function f_k at each time point for subject i. The joint distribution of $Y_i = (Y_{i1}, \ldots Y_{in_i})$ in model (10) is

$$Y_i \sim N_{n_i} \left(f_{0i}, \Sigma(\sigma^2, f_{1i}, \ldots, f_{Ki}) \right),$$

where

$$\Sigma(\sigma^2, f_{1i}, \ldots, f_{Ki}) = \sigma^2 I + \sum_{k=1}^{K} f_{ki} f_{ki}^T. \tag{14}$$

The likelihood of the model (10) is therefore

$$L(f_0, f_1, \ldots, f_K, \sigma^2) = \prod_{i=1}^{d} \phi_{n_i} \left(f_{0i}, \Sigma(\sigma^2, f_{1i}, \ldots, f_{Ki}) \right),$$

where $\phi_n(., \mu, \Sigma)$ is the probability density function of the $N_n(\mu, \Sigma)$ distribution, and the log-likelihood is

$$\ell^f(f_0, f_1, \ldots, f_K, \sigma^2) = \sum_{i=1}^{d} \log \phi_{n_i} \left(f_{0i}, \Sigma(\sigma^2, f_{1i}, \ldots, f_{Ki}) \right). \tag{15}$$

We include the superscript f to clarify that here we parameterise the process by by the functions $f_0, f_1, \ldots, f_K$, as we later consider log-likelihoods under alternative parameterisations.

To avoid over-fitting, we penalise the expected wiggliness of each subject's mean curve $\mu_i(.)$. We write $w_E = E[w(\mu_i)]$ for the expected wiggliness of μ_i, where μ_i are random variables as described by (10). There is a simple formula for the expected wiggliness:

Claim 1. *We have*

$$w_E = w(f_0) + \sum_{j=1}^{k} w(f_j).$$

All proofs are given in Appendix A. We write $w_E = w_E(f_0, f_1, \ldots, f_K)$, to make the dependence on the functions $f_0, f_1, \ldots, f_K$ explicit.

We maximise the penalised log-likelihood

$$\ell_p^f(f_0, f_1, \ldots, f_K, \sigma^2) = \ell^f(f_0, f_1, \ldots, f_K, \sigma^2) - \frac{\gamma}{2\sigma^2} w_E(f_0, f_1, \ldots, f_K), \tag{16}$$

where γ is a smoothing parameter. For now, we will focus on estimating the model parameters for fixed γ, and return to the problem of choosing γ in Section 4.7, where we also justify dividing the smoothing parameter by $2\sigma^2$.

To estimate the unknown quantities, we could choose $f_0, f_1, \ldots, f_K$ and σ^2 to maximise $\ell_p^f(f_0, f_1, \ldots, f_K)$, subject to the constraint (11), that $f_1, \ldots, f_K$ are orthogonal functions.

This very general form of the optimisation problem is not yet feasible to use in practice. It involves infinite-dimensional optimisation over all possible functions $f_0, f_1, \ldots f_K$: in Section 4.2 we will reduce this to a finite-dimensional problem by writing the f_j in terms of basis functions. A challenging constrained optimisation problem still remains, due to the orthogonality constraint on $f_1, \ldots, f_K$. In Section 4.3 we will address this by creating a new parameterisation for the problem which ensures orthogonality between $f_1, \ldots, f_K$.

4.2 Spline basis

We write

$$f_j(t) = \sum_{l=1}^{n_B} \beta_{jl} b_l(t) = \beta_j^T b(t),$$

for $j = 0, 1, \ldots, K$, where $b_l(.)$ are cubic spline basis functions, $b(t) = (b_1(t), \ldots, b_{n_B}(t))^T$ is a vector of basis functions evaluated at t, and $\beta_j = (\beta_{j1}, \ldots, \beta_{jn_B})^T$ is the vector of coefficients for f_j. In this form, the unknown quantities to estimate are $\beta_0, \beta_1, \ldots, \beta_K$ and σ^2.

The log-likelihood has the same form as (15), where the values of $f_{ki} = (f_k(t_{i1}), \ldots, f_k(t_{in_i}))^T$ depend on β_k through

$$f_k(t_{ij}) = \beta_j^T b(t_{ij}).$$

Write $X^{(i)}$ as a $n_i \times n_B$ design matrix for the basis for the ith subject, with jth row $b(t_{ij})$. Then we may write $f_{ki}(\beta_k) = X^{(i)}\beta_k$, and the log-likelihood is

$$\ell^\beta(\beta_0, \beta_1, \ldots, \beta_K, \sigma^2) = \sum_{i=1}^{d} \log \phi_{n_i}\left(f_{0i}(\beta_0), \Sigma(\sigma^2, f_{1i}(\beta_1), \ldots, f_{Ki}(\beta_K))\right).$$

We will choose our basis functions to be orthonormal, such that

$$\langle b_i, b_i \rangle = 1 \text{ and } \langle b_i, b_j \rangle = 0 \text{ for all } i \neq j.$$

We use the `orthogonalsplinebasis` R package (Redd, 2022) to do this. Claim 2 says that we can obtain a set of orthogonal functions from an orthonormal basis by choosing a set of orthogonal coefficient vectors.

Claim 2. *Suppose that $b_1, \ldots, b_{n_B}$ are an orthonormal basis, and $f_j(t) = \sum_{l=1}^{n_B} \beta_{jl} b_l(t)$. Then the functions $f_1, \ldots f_K$ are orthogonal if and only if the coefficient vectors $\beta_1, \ldots, \beta_K$ are orthogonal, that is, if*

$$\langle \beta_i, \beta_j \rangle = 0, \text{ for all } i \neq j \in \{1, \ldots, K\} \tag{17}$$

where $\langle \beta_i, \beta_j \rangle = \beta_i^T \beta_j$ is the usual inner product for vectors.

Finding the wiggliness of each f_j from the spline basis is straightforward: there is a matrix S such that $w(f_j) = \beta_j^T S \beta_j$. Given the coefficient vectors $\beta_0, \beta_1, \ldots, \beta_K$, the expected wiggliness may be written

$$w_E = \sum_{j=0}^{K} \beta_j^T S \beta_j.$$

We will write $w_E = w_E(\beta_0, \beta_1 \ldots, \beta_K)$ to make the dependence on the coefficient vectors explicit.

We could therefore estimate the unknown parameters $\beta_0, \beta_1, \ldots, \beta_K$ and σ^2 by maximising the penalised log-likelihood

$$\ell_p^\beta(\beta_0, \beta_1, \ldots, \beta_K, \sigma^2) = \ell^\beta(\beta_0, \beta_1, \ldots, \beta_K, \sigma^2) - \frac{\gamma}{2\sigma^2} w_E(\beta_0, \beta_1, \ldots, \beta_K) \tag{18}$$

subject to the constraint that $\beta_1, \ldots, \beta_K$ are orthogonal vectors. The constraint makes this a challenging optimisation problem, and in Section 4.3 we will develop a new parameterisation which ensures the component vectors $\beta_1, \ldots, \beta_K$ are orthogonal, to remove the constraint at the optimisation stage.

4.3 Orthogonality transform

In this section we describe a reparameterisation $\alpha_1, \ldots, \alpha_K$ such that any orthogonal set of vectors $\beta_1, \ldots, \beta_K$ may be obtained for some choice of the α_ks. In combination with the orthonormal spline basis from Section 4.2, this will enable us to parameterise the set of orthogonal functions $f_1, \ldots, f_k$, without the need to place any constraints on the values of the new parameters $\alpha_1, \ldots, \alpha_K$.

We do not place any constraints on β_1, and let $\beta_1 = \alpha_1$, for any $\alpha_1 \in \mathbb{R}^{n_B}$. For each $k > 1$, we need to ensure that β_k is orthogonal to each of $\beta_1, \ldots, \beta_{k-1}$, leading to $k - 1$ constraints, so we

take $\alpha_k \in \mathbb{R}^{n_B - k + 1}$. We may rewrite the constraint that β_k is orthogonal to each of $\beta_1, \ldots, \beta_{k-1}$ in matrix-vector form, as

$$B_{k-1}^T \beta_k = 0, \tag{19}$$

where B_{k-1} is the $n_B \times (k-1)$ matrix with columns columns $\beta_1, \ldots, \beta_{k-1}$. To find β_k from α_K and $\beta_1, \ldots, \beta_{k-1}$, we follow a general recipe described in Section 1.8.1 of Wood (2017):

1. Find a QR decomposition of B_{k-1}, as $B_{k-1} = Q_{k-1} R_{k-1}$, where Q_{k-1} is an orthogonal matrix and R_{k-1} is upper triangular.

2. Partition $Q_{k-1} = (S_{k-1} : T_{k-1})$ so that T_{k-1} contains the final $n_B - k + 1$ columns of Q_{k-1}.

3. Let

$$\beta_k = T_{k-1} \alpha_k.$$

For any $\alpha_k \in \mathbb{R}^{n_B - k + 1}$, this give a β_k meeting the orthogonality constraint (19).

We may find the penalised likelihood under our new parameterisation by first finding $\beta = (\beta_1, \ldots, \beta_K)$ from $\alpha = (\alpha_1, \ldots, \alpha_k)$ then substituting into the penalised log-likelihood (18) for $(\beta_0, \beta, \sigma^2)$, to give

$$\ell_p^\alpha(\beta_0, \alpha, \sigma^2) = \ell_p^\beta(\beta_0, \beta(\alpha), \sigma^2).$$

For numerical stability, we maximise the penalised log-likelihood over $\log \sigma$ rather than σ^2. The full set of parameters is $\theta = (\beta_0, \alpha, \log \sigma)$, with penalised log-likelihood

$$\ell_p(\theta) = \ell_p(\beta_0, \alpha, \log \sigma) = \ell_p^\alpha(\beta_0, \alpha, \exp(2 \log \sigma)). \tag{20}$$

4.4 Maximising the penalised log-likelihood

We find the maximum penalised likelihood estimate $\hat{\theta}$ by maximising $\ell_p(\theta)$ from (20) over θ. In practice, we do this by first fitting the model with $K = 0$, then increasing K one at a time, using the parameter values from the previous fit to give starting values for the optimisation.

We use the BFGS method to maximise the penalised log-likelihood. For speed, it is important to have access to the gradient of the penalised log-likelihood, which would be difficult to find by hand because of the complexity of the transformation in 4.3. We used automatic differentiation to obtain the gradient, by using the Stan Math C++ library in R (Stan Development Team, 2020).

4.5 Estimating the subject-specific mean curves

Given the maximum penalised likelihood estimate $\hat{\theta}$, we can estimate the subject-specific mean curves as

$$\hat{\mu}_i(t) = \hat{f}_0(t) + \sum_{k=1}^K \hat{u}_{ki} \hat{f}_k(t),$$

where $\hat{f}_k(t) = \hat{\beta}_k^T b(t)$, where $\hat{\beta}_k = \beta_k(\hat{\theta})$, and we estimate $u_i = (u_{i1}, \ldots, u_{iK})^T$ by

$$\hat{u}_i = \hat{\Sigma}_i^{-1}(y_i - \hat{f}_{0i}),$$

where $\hat{\Sigma}_i = \Sigma(\hat{\sigma}^2, \hat{f}_{1i}, \ldots, \hat{f}_{Ki})$ from (14).

4.6 Confidence intervals

We express the uncertainty in our estimated mean curves $\hat{\mu}_i(t)$ through pointwise confidence intervals.

We may always express each subject-specific mean in terms of the fixed set of basis functions $b(.)$. For fixed β and u, we may write

$$\mu_i(t) = \beta_0^T b(t) + \sum_{k=1}^K u_{ik} \beta_k^T b(t) = (\beta_0 + \sum_{k=1}^K u_{ik} \beta_k)^T b(t) = \delta_i^T b(t)$$

where

$$\delta_i = \beta_0 + \sum_{k=1}^{K} u_{ik} \beta_k$$

are the basis coefficients for μ_i, which depend on β and u_i.

To find a pointwise confidence interval for $\mu_i(t)$, we use a parametric bootstrap approach. We will first find a parametric bootstrap sample $\{\delta_i^{(1)}, \ldots, \delta_i^{(n_S)}\}$ for the basis coefficients δ_i of the subject-specific mean functions.

For the fitted model (with K and γ treated as fixed at their chosen values), we first find the Hessian matrix H of the penalised log-likelihood at the maximum penalised Lakewood estimate $\hat{\theta}$, and find $V = -H^{-1}$. For each $j = 1, \ldots, n_s$, to find the sample $\delta^{(j)}$, we:

1. Generate $\theta^{(j)} \sim N_p(\hat{\theta}, V)$.

2. Reparameterise out of the orthogonal parameterisation, to find $\beta_k^{(j)} = \beta_k(\theta^{(j)})$. Write $f_k^{(j)}(t) = [\beta_k^{(j)}]^T b(t)$.

3. For each subject $i = 1, \ldots, d$:

   (a) Find $f_{0i}^{(j)}, f_{1i}^{(j)}, \ldots, f_{Ki}^{(j)}$, where $f_{ki} = (f_k^{(j)}(t_{i1}), \ldots, f_k^{(j)}(t_{in_i}))^T$, is the vector which evaluates the function $f_k^{(j)}$ at each time point for subject i.

   (b) Find $\Sigma_i^{(j)} = \Sigma([\sigma^2]^{(j)}, f_{1i}^{(j)}, \ldots, f_{Ki}^{(j)})$ using (14), and $\hat{u}_i^{(j)} = (\Sigma_i^{(j)})^{-1}(y_i - f_{0i}^{(j)})$.

   (c) Sample $u_i^{(j)} | \theta^{(j)}, y, \sim N(\hat{u}_i^{(j)}, \Sigma_i^{(j)})$.

   (d) Find

   $$\delta_i^{(j)} = \beta_0^{(j)} + \sum_{k=1}^{K} u_{ik}^{(j)} \beta_k^{(j)}$$

4. Return $\delta^{(j)} = (\delta_1^{(j)}, \ldots, \delta_d^{(j)})$.

Given $\{\delta_i^{(1)}, \ldots, \delta_i^{(n_S)}\}$, we may then find a confidence interval for $\mu_i(t)$ (at any time point t) by finding

$$[\mu_i(t)]^{(j)} = \delta_i^{(j)} b(t)$$

for each sample $j = 1, \ldots, n_S$, then using appropriate quantiles of $\{[\mu_i(t)]^{(1)}, \ldots, [\mu_i(t)]^{(n_S)}\}$ (e.g. the 2.5% and 97.5% quantiles for a 95% interval).

In addition to confidence intervals about μ_i, we could also use the sample $\delta^{(j)}$ to find confidence intervals for other quantities of interest. For instance, we can find confidence intervals for the derivative of the subject-specific mean function by finding

$$[\mu_i'(t)]^{(j)} = \delta_i^{(j)} b'(t).$$

This process dues not allow for uncertainty in the tuning parameters K and γ, so there may be some potential for under-coverage. However, we will see that the confidence intervals have close to nominal coverage in simulation studies.

4.7 Choosing the tuning parameters

4.7.1 Choosing the number of functional principal components K

For a fixed smoothing parameter γ, we find that the fit eventually stabilises as we increase the number of functional principal components K, with with $\hat{\lambda}_K = \|\hat{f}_K\|^2$ approaching zero. We could take K very large, and still get a good fit, but with lots of unimportant functional principal components, with $\hat{\lambda}_k \approx 0$. For computational reasons, we prefer K to choose as small as possible while including all of the important functional principal components.

To do this, we use the fraction of variance explained. For each candidate K, we estimate the error variance $\hat{\sigma}_K^2$. There is a residual variance which cannot be explained by variation in the

subject-specific mean functions, which we write as $\hat{\sigma}_\infty^2 = \lim_{K\to\infty} \hat{\sigma}_K^2$. The non-residual variance for each K is $\hat{\sigma}_K^2 - \hat{\sigma}_\infty^2$. The fraction of variance explained is

$$\text{FVE}(K) = \frac{\hat{\sigma}_0^2 - \hat{\sigma}_K^2}{\hat{\sigma}_0^2 - \hat{\sigma}_\infty^2}$$

We aim to choose the smallest K so that $\text{FVE}(K) > t_{\text{FVE}}$, for some threshold t_{FVE}, close to 1. In all the later examples we use $t_{\text{FVE}} = 0.999$.

We cannot calculate $\hat{\sigma}_\infty^2$, since we do not fit the model with infinitely large K. For sufficiently large K, we will have $\hat{\sigma}_K^2 \approx \hat{\sigma}_\infty^2$. Starting with $K_{\max} = 2$, fit the model for all $K \le K_{\max}$, and use $\hat{\sigma}_{K_{\max}}^2$ in place of $\hat{\sigma}_\infty^2$, to give

$$\text{FVE}(K; K_{\max}) = \frac{\hat{\sigma}_0^2 - \hat{\sigma}_K^2}{\hat{\sigma}_0^2 - \hat{\sigma}_{K_{\max}}^2}.$$

We check whether $\text{FVE}(K_{\max} - 1; K_{\max}) > t_{\text{FVE}}$, and choose $K = K_{\max} - 1$ if so. If not, we increase $K_{\max}$ by one, refit the model with $K = K_{\max}$, and repeat the process until we have a sufficiently large fraction of variance explained.

4.7.2 Choosing the smoothing parameter γ

We choose the smoothing parameter γ to maximise an approximate marginal likelihood. To do this, we must first consider a Bayesian interpretation of the penalty term.

We may rewrite the penalty term in (18) as

$$-\frac{\gamma}{2\sigma^2} w_E = \sum_{j=0}^{K} -\frac{\gamma}{2\sigma^2} \beta_j^T S \beta_j.$$

We use the equivalence between penalised likelihood estimation with penalty

$$-\frac{\gamma}{2\sigma^2} \beta_k^T S \beta_k$$

and maximum a posteriori estimation with improper Bayesian prior

$$\beta_k \sim N\left(0, \frac{\sigma^2}{\gamma} S^-\right), \tag{21}$$

an improper normal prior where S^- is the pseudo-inverse of the matrix S, since S is not of full rank (Wood, 2017, Section 4.2.4).

The prior for β_0 is given by (21). Since we reparameterise $\beta = (\beta_1, \ldots, \beta_K)$ in terms of $\alpha = (\alpha_1, \ldots, \alpha_K)$, we seek a prior for α such that if α has this distribution then $\beta_k = \beta_k(\alpha)$ has distribution (21) for $k = 1, \ldots, K$.

We may achieve this by letting

$$\alpha_1 \sim N\left(0, \frac{\sigma^2}{\gamma} S^-\right)$$

and

$$\alpha_k | \alpha_1, \ldots, \alpha_{k-1} \sim N\left(0, \frac{\sigma^2}{\gamma} S_k^-\right), \quad \text{for } k = 2, \ldots, K,$$

where

$$S_k = T_{k-1}^T S T_{k-1}.$$

The matrices T_{k-1} are defined in step (2) of the transformation in Section 4.3, and depend on $\alpha_1, \ldots, \alpha_{k-1}$.

Since we optimise our penalised log-likelihood over $\theta = (\beta_0, \alpha_1, \ldots, \alpha_K, \log \sigma)^T$, we will also include a prior on $\log \sigma$, which we describe shortly. Our penalty over θ is

$$\text{pen}(\theta | \gamma) = -\frac{\gamma}{2\sigma^2} \beta_0^T S \beta_0 - \sum_{k=1}^{K} \alpha_k S_k^T \alpha_k$$

13

We then rewrite the log prior as the penalty plus a remainder term, as

$$\log \pi(\theta|\gamma) = \text{pen}(\theta|\gamma) + r(\theta, \gamma).$$

If we choose the improper prior $\pi(\log \sigma) \propto \sigma^{-2R}$ for $\log \sigma$, where $R = \sum_{k=0}^{K} r_k$ and $r_k = \text{rank}(S_k) = \min\{n_B - 2, n_B - k + 1\}$, then the remainder term is

$$r(\theta, \gamma) = \frac{1}{2} \sum_{k=0}^{K} \log |S_k|_+ + R(\log \gamma - 1).$$

At first sight, it appears that there is no dependence on θ in the remainder $r(\theta, \gamma)$, which would lead to an exact match between the maximum a-posteriori estimate and the penalised likelihood estimate. However, there is a small dependence on α, since S_K depends on $\alpha_1, \ldots, \alpha_{k-1}$, so the maximum a-posteriori estimate will differ slightly from the penalised likelihood estimate.

We choose γ to maximise a quantity based on the marginal likelihood for γ,

$$c_\gamma = \int \pi_{\text{unnorm}}(\theta, \gamma|y) d\theta \tag{22}$$

where

$$\pi_{\text{unnorm}}(\theta, \gamma|y) = f(y|\theta)\pi(\theta|\gamma).$$

Making a Laplace approximation to (22) gives an approximate log-marginal likelihood

$$\log \hat{c}_\gamma = \log \pi_{\text{unnorm}}(\theta_{\text{MAP}}(\gamma), \gamma) + \frac{p}{2} \log(2\pi) - \frac{1}{2} \big| \log H_{\text{MAP}}(\gamma) \big|, \tag{23}$$

where $\theta_{\text{MAP}}(\gamma)$ is the maximum a-posteriori estimate for θ given γ, $H_{\text{MAP}}(\gamma)$ is the Hessian of $\log \pi_{\text{unnorm}}(\theta|y)$ at $\theta_{\text{MAP}}(\gamma)$ and $p = \dim(\theta)$.

We find a further approximation to the marginal likelihood by replacing $\theta_{\text{MAP}}(\gamma)$ by the penalised likelihood estimate $\hat{\theta}(\gamma)$ in (23), also replacing $H_{\text{MAP}}(\gamma)$ by $\hat{H}(\gamma)$, the Hessian of the penalised log-likelihood at its maximum, which gives our approximate log marginal likelihood criterion

$$\log \tilde{c}_\gamma = \log \pi_{\text{unnorm}}(\hat{\theta}(\gamma), \gamma) + \frac{p}{2} \log(2\pi) - \frac{1}{2} \big| \log \hat{H}(\gamma) \big|,$$

$$= \ell_p(\hat{\theta}(\gamma), \gamma) + r(\hat{\theta}(\gamma), \gamma) + \frac{p}{2} \log(2\pi) - \frac{1}{2} \big| \log \hat{H}(\gamma) \big|. \tag{24}$$

In examples, we find that maximising this criterion tends to give an appropriate value of γ. The inference is typically not sensitive to the precise value of γ. Further study of the properties of the criterion would be useful, and is left for future work.

In theory, the same marginal likelihood criterion (24) could be optimised over K as well as γ. However, computing the marginal likelihood criterion requires us to first find the Hessian matrix at the maximum of the penalised log-likelihood, which is relatively computationally expensive. We reduce the computational burden by finding a marginal likelihood just once for each γ, choosing K given γ as described in Section 4.7.1.

5   Simulation studies

5.1   General setup

5.1.1   Introduction

We compare the penalised likelihood approach against existing approaches through simulation studies. We simulate from model (1), with a wide range of different choices for the true subject-specific mean functions μ_i. These mean functions are of the form (10), for some number of functional principal components $K = K_0$. The various processes used to generate the data are described in sections 5.2–5.4. For each choice, we generate 100 datasets, then use a range of methods to estimate the subject-specific mean curves μ_i. A common set of methods are used for estimation in each case, as described in section 5.1.2. The metrics we use to compare the quality of inference from each method are described in section 5.1.3.

5.1.2 Estimation methods

For ease of reference later on, we give short names to our chosen estimation methods (highlighted below in bold type). The methods used for estimating the subject-specific mean curves μ_i are:

1. **RI**: The simple random intercept model (3).

2. **RS**: The simple random slopes model (4).

3. PACE (Yao et al., 2005), through the `FPCA` function in the `fdapace` R package. To choose K, we either fix $K = K_0$ (**PACE-Oracle**, not realistic in practice) or estimate K, either by making the fraction of variance explained (FVE) at least 0.95 (**PACE-95**) or 0.99 (**PACE-99**) , or using AIC (**PACE-AIC**) or BIC (**PACE-BIC**). We use generalised cross validation to select bandwidths for smoothing the mean and covariance functions (as this gave better results than the default choice), and default values for other parameters.

4. The method of Di et al. (2009), through the `fpca.sc` function in the `refund` R package. To choose K, we either fix $K = K_0$ (**Di-Oracle**) or estimate K to make the FVE at least 0.95 (**Di-95**) or 0.99 (**Di-99**). We use default values for other parameters.

5. **PL**: the proposed penalised likelihood method. We use the methods for choosing K and γ described in Section 4.7, with $t_{\text{FVE}} = 0.999$ for the FVE threshold.

We also used the method of Goldsmith et al. (2013), through the `ccb.fpc` function in `refund`, in some preliminary simulation runs, but it appeared to give worse estimates than the Di methods in the cases we considered, with extremely large confidence intervals. Since this method was also more computationally expensive than the others, we decided not to include it in the full simulation studies.

5.1.3 Measures used for comparisons

We compare the quality of inference from the different methods by using the mean integrated squared error for each estimate of the subject specific mean functions $\hat{\mu} = (\hat{\mu}_1, \ldots, \hat{\mu}_d)$, across the range of times observed in the data. We define the integrated squared error for subject i as

$$\text{ISE}(\hat{\mu}_i) = \int_{t_{\min}}^{t_{\max}} \left[\hat{\mu}_i(t) - \mu_i(t)\right]^2 dt,$$

where $t_{\min} = \min_{i,j}\{t_{ij}\}$ and $t_{\max} = \max_{i,j}\{t_{ij}\}$. The mean integrated squared error is

$$\text{MISE}(\hat{\mu}) = \frac{1}{d} \sum_{i=1}^{d} \text{ISE}(\hat{\mu}_i).$$

We then find the MISE for each of the 100 datasets generated from each process, and report the RMISE – the square root of the mean MISE across datasets – for each method.

In some cases, it became clear that the MISE of Di and PACE estimators were sometimes large because of a few subjects with very large error. To understand the quality of inference for an average subject, we decided to also look at the median integrated squared error

$$\text{MedISE}(\hat{\mu}) = \text{median}\big(\text{ISE}(\hat{\mu}_1), \ldots, \text{ISE}(\hat{\mu}_d)\big).$$

We find the MedISE for each of the 100 datasets generated from each process, then report the RMedISE – the square root of the median MedISE across datasets – for each method.

To give a clear comparison of the performance of the new PL estimator relative to existing methods, we will consider the relative RMedISE of each estimator $\hat{\mu}$ compared with the PL estimator,

$$\text{Relative RMedISE}(\hat{\mu}) = \frac{\text{RMedISE}(\hat{\mu})}{\text{RMedISE}(\hat{\mu}_{\text{PL}})}.$$

Figure 6: Relative root median integrated square error for each estimator relative to the PL estimator, for each subcase of the random intercept simulations.

For each case, it is useful to be able to examine the various estimated subject-specific mean functions for a typical simulation run and subject. To find a run and subject with typical performance, we first choose the run with median integrated squared error closest to the overall MedISE for PACE-Oracle, and within that run find the subject with integrated squared error for PACE-Oracle closest to the median integrated square error.

For the Di and PL methods, we compute pointwise 95% confidence intervals associated with each estimated subject-specific mean $\hat{\mu}_i$. We also attempted to compute confidence intervals from PACE, but this resulted in errors on a substantial number of runs, so the comparison is not included here. For each dataset, we compute the average pointwise coverage of these intervals across the range of times observed in the data $[t_{\min}, t_{\max}]$. For each method, we report the average coverage across the datasets.

5.2 Random intercept model

We simulate data from a random intercept model

$$y_{ij} = \beta_0 + \beta_1 t_{ij} + u_{0i} + \epsilon_{ij}, \quad i = 1, \ldots d, \quad j = 1, \ldots, n_i,$$

where $u_{0i} \sim N(0, \sigma_u^2)$ and $\epsilon_{ij} \sim N(0, \sigma^2)$. We generate data from 16 processes of this form, given by all possible combinations of: $\beta_0 = -1$, $\beta_1 = 2$, $\sigma_u \in \{0.5, 1\}$, $\sigma \in \{0.1, 0.5\}$, $d \in \{20, 50\}$, $n_i \in \{5, 10\}$ (with an equal number of observations on each subject). The time points t_{ij} are uniformly distributed on $[0, 1]$.

Tables 1 and 2 in Appendix B show the RMISE and RMedISE for each case and method. Figure 6 plots the relative RMedISE for each case, showing only Oracle methods for PACE and Di, with K fixed at 1. In reality, we must estimate K, which increases the error further. The PACE and Di estimators always have the largest errors, several times larger than the PL estimator in many cases. The loss of efficiency by using the PL method relative to the true RI model is small: the relative error for the RI model is always at least 0.9.
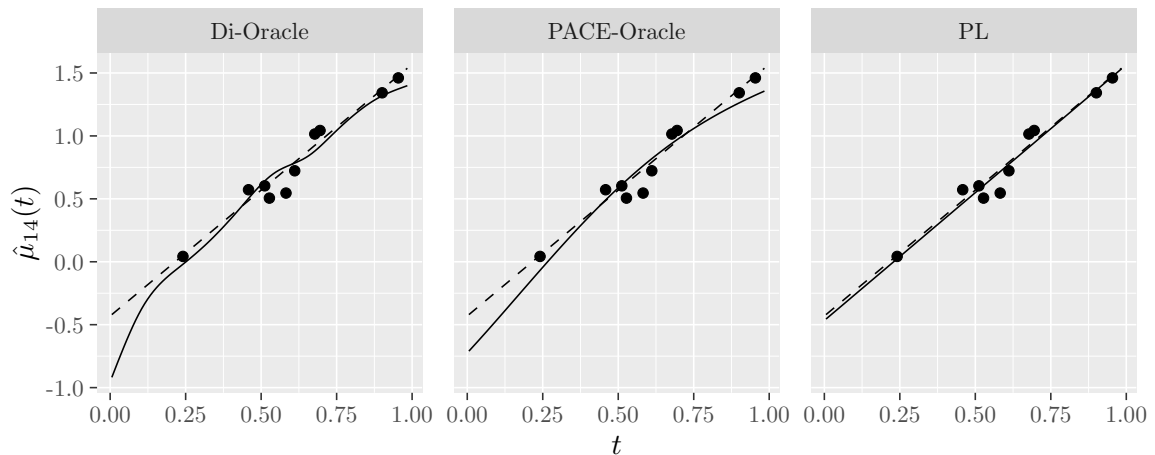
Figure 7: Estimated mean curves for a typical simulation run (67) and subject (14) in the random intercept simulations with $n_i = 10$, $d = 20$, $\sigma = 0.1$, $\sigma_u = 1$.

The PACE and Di estimators perform worst when the error variance is small ($\sigma = 0.1$) and the random effects variance is large ($\sigma_u = 1$), with RMedISE 4–7 times larger than PL in these cases (similarly, the RMISE is 5–8 times larger than for PL).

We examine the case $n_i = 10$, $d = 20$, $\sigma = 0.1$, $\sigma_u = 1$ in more detail. Figure 7 shows various estimated mean curves for a typical simulation run (67) and subject (14), where the run and subject were chosen as described in 5.1.3. The PL estimate matches the true linear mean function closely, while the Di and PACE estimates do not give a fit close to a straight line.

Table 3 gives the pointwise coverage of nominally 95% confidence intervals for the subject-specific mean curves, for the PL and Di methods. The PL confidence intervals have approximately correct coverage, whereas the Di confidence intervals all have much lower than the nominal coverage.

5.3 Random slopes model

We simulate data from a random slopes model

$$y_{ij} = \beta_0 + \beta_1 t_{ij} + u_{0i} + u_{1i} + \epsilon_{ij}, \quad i = 1, \ldots d, \quad j = 1, \ldots, n_i,$$

where $u_i = (u_{0i}, u_{1i})^T \sim N_2(0, \Sigma_u)$, $\epsilon_{ij} \sim N(0, \sigma^2)$, and

$$\Sigma_u = \begin{pmatrix} \sigma_0^2 & \rho\sigma_0\sigma_1 \\ \rho\sigma_0\sigma_1 & \sigma_1^2 \end{pmatrix}.$$

We generate data from 32 processes of this form, given by all possible combinations of: $\beta_0 = -1$, $\beta_1 = 2$, $\sigma_0 \in \{0.5, 1\}$, $\sigma_1 = 0.5$, $\rho \in \{0, 0.5\}$, $\sigma \in \{0.1, 0.5\}$, $d \in \{20, 50\}$, $n_i \in \{5, 10\}$ (with an equal number of observations on each subject). The time points t_{ij} are uniformly distributed on $[0, 1]$.

Tables 4 and 5 in Appendix B show the RMISE and RMedISE for each case and method. The random slopes model and new PL method give the lowest errors in each case. In many cases, the PACE and Di estimators have errors several times larger than the the PL estimator.

Figure 8 plots the relative RMedISE for each case, showing only Oracle methods for PACE and Di, with K fixed at 2. The PACE and Di estimators always have the largest errors, several times larger than PL in many cases. The loss of efficiency by using the PL estimator relative to the true RS model is small: the relative error for the RS model is always at least 0.95.

The PACE and Di estimators perform worst when the error variance is small ($\sigma = 0.1$) and the random effects variance is large ($\sigma_u = 1$), with similar performance whether the random intercept

17
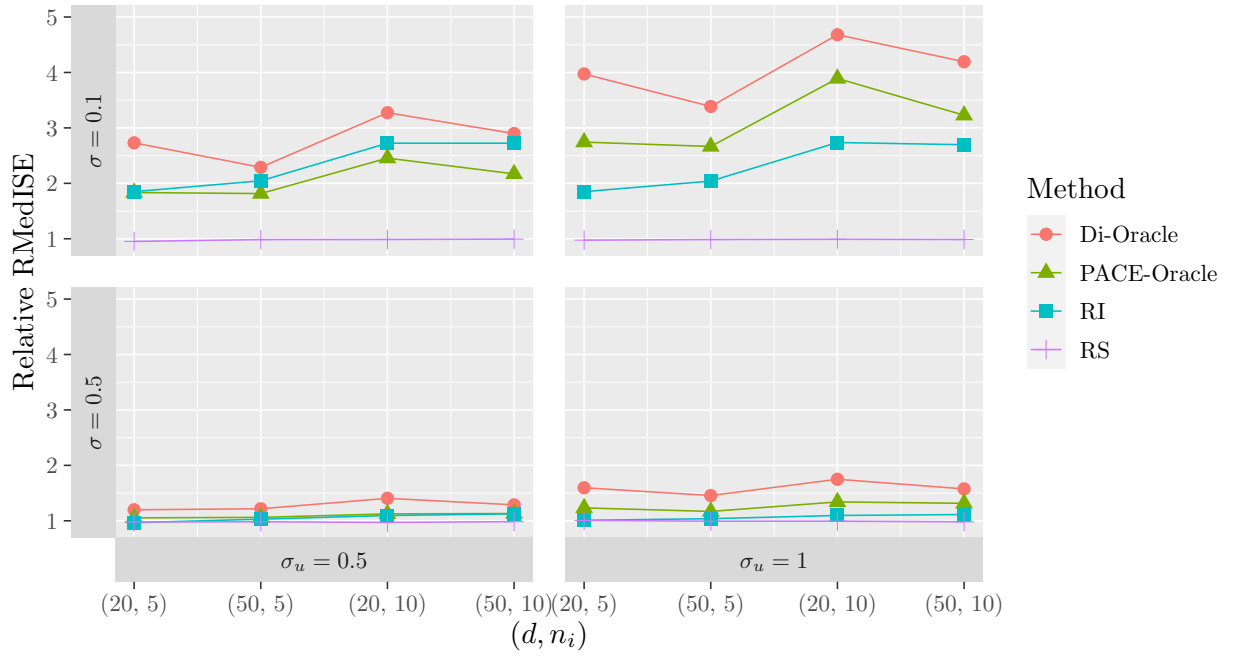
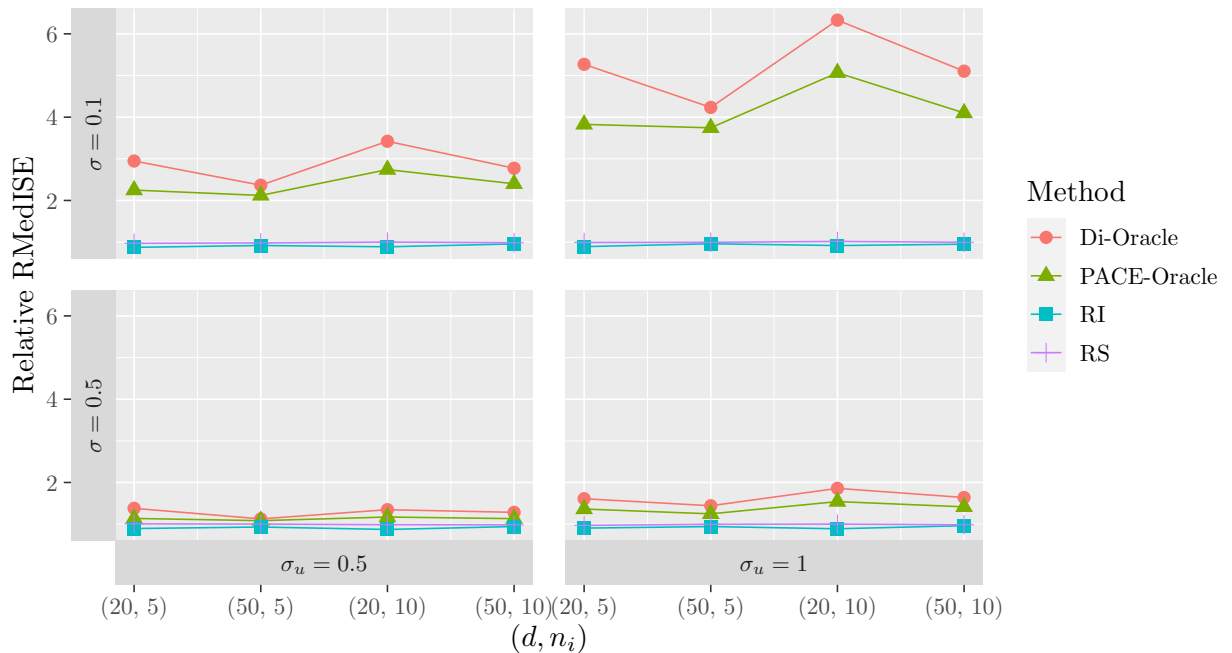(a) $\rho = 0$
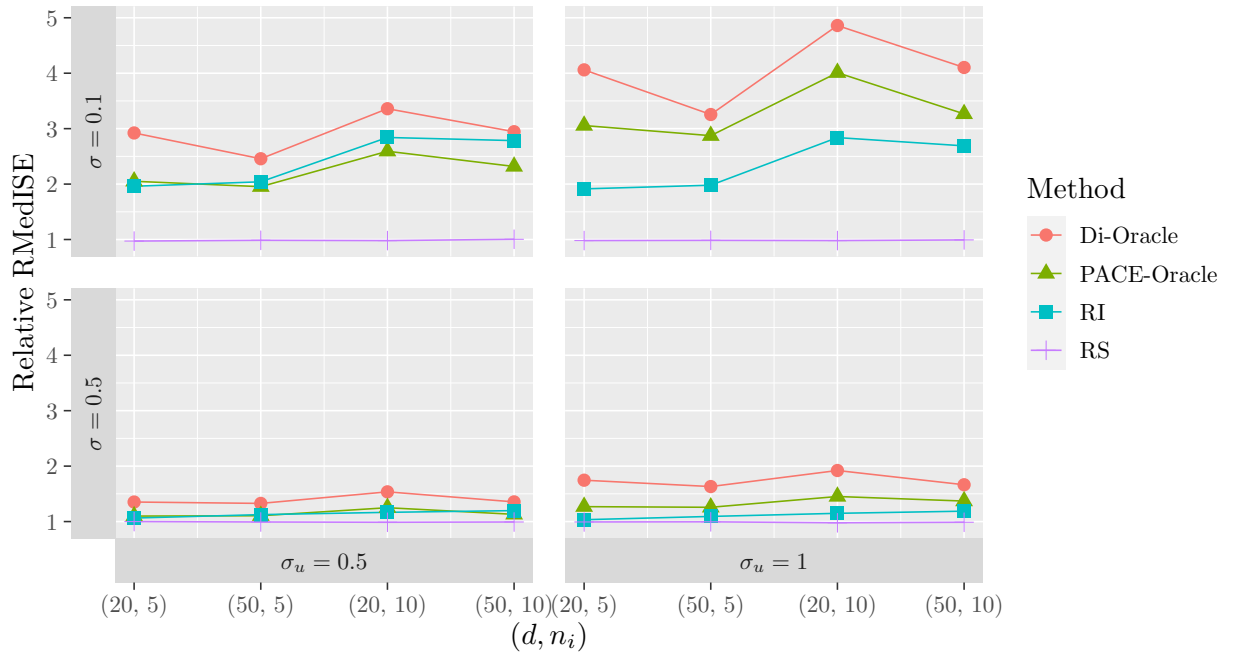
(b) $\rho = 0.5$

Figure 8: Relative relative median integrated square error for each estimator relative to the PL estimator, for each subcase of the random slope simulations.
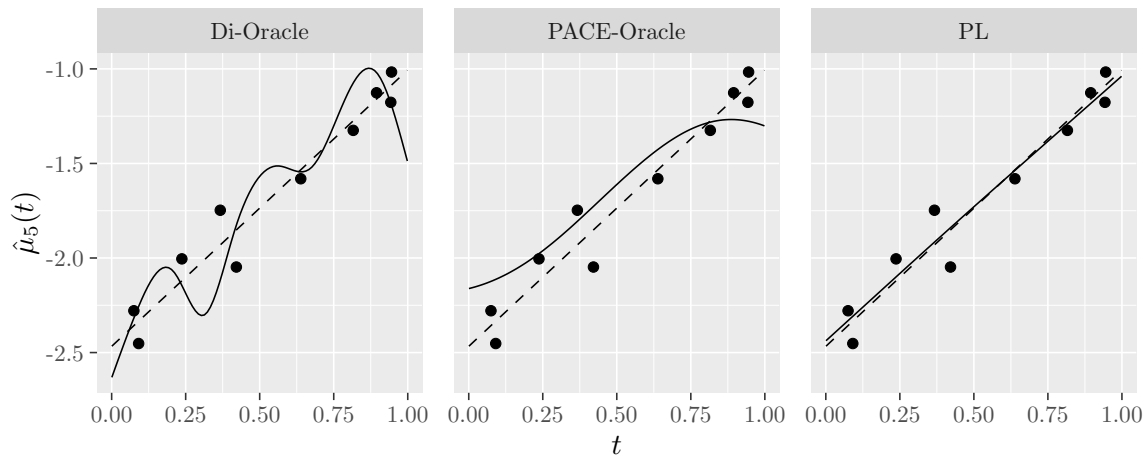
Figure 9: Estimated mean curves for a typical simulation run (79) and subject (5) in the random slope simulations with $n_i = 10$, $d = 20$, $\sigma = 0.1$, $\sigma_u = 1$, $\rho = 0.5$.

and slope are uncorrelated ($\rho = 0$) or correlated ($\rho = 0.5$). In many cases, despite the presence of random slopes, the RI estimator outperforms the PACE and Di estimators.

We examine the case $n_i = 10$, $d = 20$, $\sigma = 0.1$, $\sigma_u = 1$, $\rho = 0.5$ in more detail. Figure 9 shows various estimated mean curves for a typical simulation run (79) and subject (5), where the run and subject were chosen as described in 5.1.3. As in the random intercept simulations, the PL estimate matches the true linear mean function closely, while the Di and PACE estimates do not give a fit close to a straight line.

Table 6 gives the pointwise coverage of nominally 95% confidence intervals for the subject-specific mean curves, for the PL and Di methods. As before, the PL confidence intervals have approximately correct coverage, whereas the Di confidence intervals all have much lower than the nominal coverage.

5.4 One-dimensional variation

We simulate from the process used in Figure 5, which has one-dimensional variation, but is not a random intercept model. The data are generated by

$$y_{ij} = s(t)(\beta_0 + \beta_1 t_{ij} + u_i) + \epsilon_{ij}, \quad i = 1, \ldots d, \quad j = 1, \ldots, n_i,$$

where $u_i \sim N(0, \sigma_u^2)$, $\epsilon_{ij} \sim N(0, \sigma^2)$, and

$$s(t) = [0.5 + 0.1(10t - 5)^2]^{-1}.$$

We generate data from 16 processes of this form, given by all possible combinations of: $\beta_0 = -0.5$, $\beta_1 = 0.1$, $\sigma_u \in \{0.5, 1\}$, $\sigma \in \{0.1, 0.5\}$, $d \in \{20, 50\}$, $n_i \in \{5, 10\}$

Tables 7 and 8 in Appendix B show the RMISE and RMedISE for each case and method. In many cases, all other estimators have error several times larger than the the PL estimator. Figure 10 plots the relative RMedISE for each case, showing only Oracle methods for PACE and Di, with K fixed at 1. The PACE and Di estimators always perform worse than the PL estimator, with error several times larger than the PL estimator in many cases. Unsurprisingly, the RI and RS estimators do badly here, as they model the subject-specific mean curves as straight lines, which is far from the truth.

As in other cases, the PACE and Di estimators perform worst when the error variance is small ($\sigma = 0.1$) and the random effects variance is large ($\sigma_u = 1$). We examine the case $n_i = 10$, $d = 20$, $\sigma = 0.1$, $\sigma_u = 1$ in more detail. Figure 11 shows various estimated mean curves for a

19
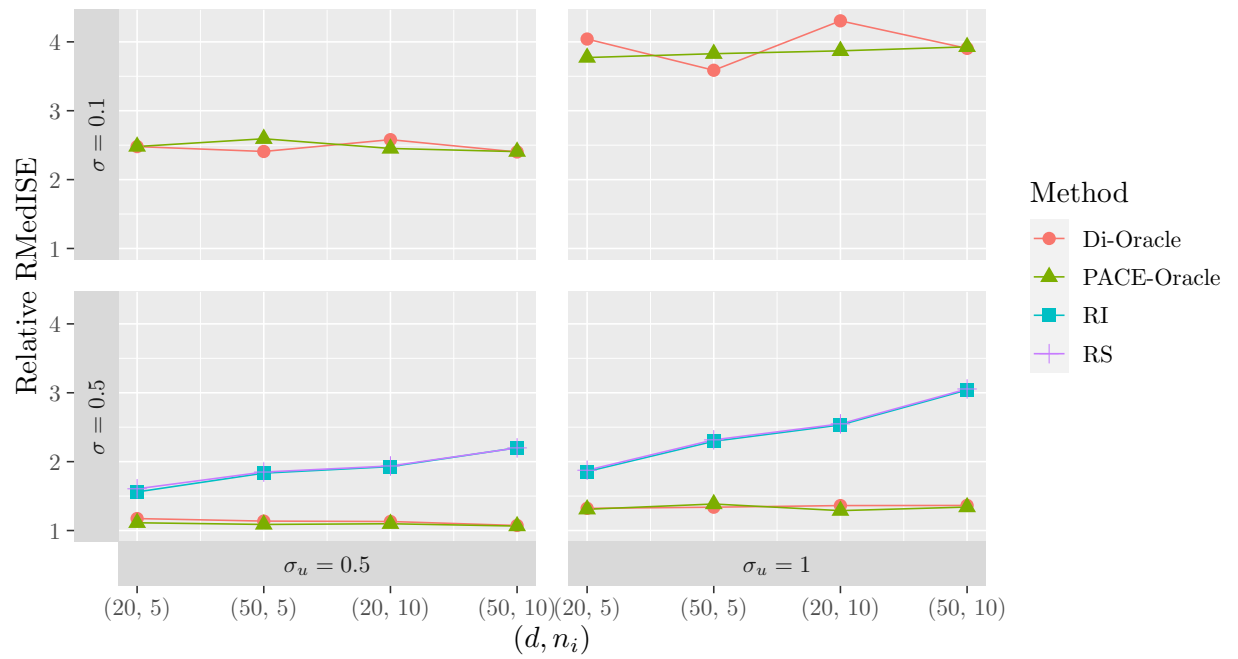
Figure 10: Relative root median integrated square error for each estimator relative to the PL estimator, for each subcase of the one-dimensional variation simulations. The y axis is truncated at 4.3: in the $\sigma = 0.1$ cases the relative error for RI and RS exceeded this threshold, and are not plotted.
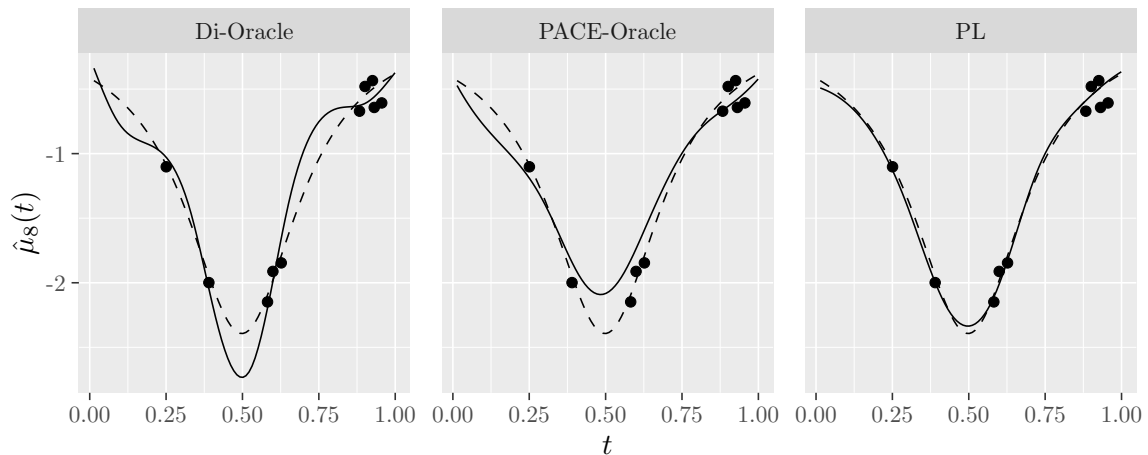
Figure 11: Estimated mean curves for a typical simulation run (71) and subject (14) in the one-dimensional variation simulations with $n_i = 10$, $d = 20$, $\sigma = 0.1$, $\sigma_u = 1$.

typical simulation run (71) and subject (8), where the run and subject were chosen as described in 5.1.3. While the Di and PACE estimates match the overall shape reasonably well, they have substantially larger errors than the PL estimate.

Table 9 gives the pointwise coverage of nominally 95% confidence intervals for the subject-specific mean curves, for the PL and Di methods. As in the previous examples, the PL confidence intervals have approximately correct coverage, whereas the Di confidence intervals all have much lower than the nominal coverage.

6 Application: percent body fat in adolescent girls

We return to the data from Section 2.1, and estimate the subject-specific mean curves using the new approach. We choose $K = 4$ components, and estimate the error standard deviation as $\hat{\sigma} = 0.32$.

The `fat` data is provided by the ALA R package, subject to the note that the data "represent a subset of the study materials and should not be used to draw substantive conclusions". We use this data to a demonstrate methodology, and the conclusions we make should not be treated as substantive scientific conclusions.

Figure 12 shows estimated percent body fat $\hat{\mu}_i(t)$ for the first twenty girls, with 95% confidence intervals. The fit appears reasonable, but some work is needed to generalise beyond individuals, to understand how percent fat body varies with time in the population as whole.

Figure 13 shows estimates and confidence interval for the population averaged mean curve. The population averaged mean curve from the piecewise linear model from Fitzmaurice et al. (2011) is overlaid. From this, it is clear that there is variation in the population averaged percent fat beyond this piecewise linear model. On average, there is relatively little change in percent body fat until around 6 months before menarche. From there until around 2 years after menarche there is a rapid increase in average percent body fat. From 2 to 4 years after menarche, percent body fat is still increasing on average, but at a slower rate.

The population averaged percent body fat doesn't tell us anything about variation between subjects. We might be interested in such variation, as well as the average behaviour. Figure 14 summarises the diversity of estimated rates of fat growth between different subjects, showing the median estimated rate of fat growth, and bands covering the central 50% and 90% of subjects.

Nearly all subjects are estimated to have increasing percent body fat from shortly before menarche until 2 years after, with rapid growth (at least 2 percentage points per year) around 6
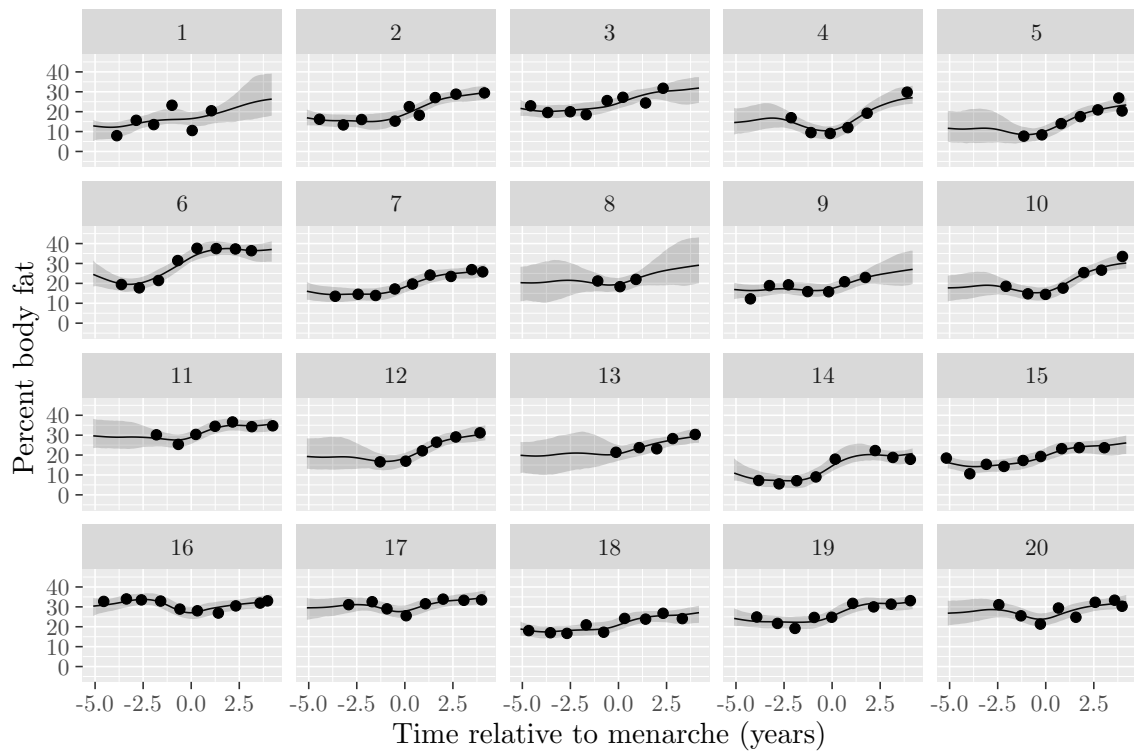
Figure 12: Percentage body fat against time, for the first twenty girls in the `fat` data, with fitted curves and 95% confidence intervals overlaid.
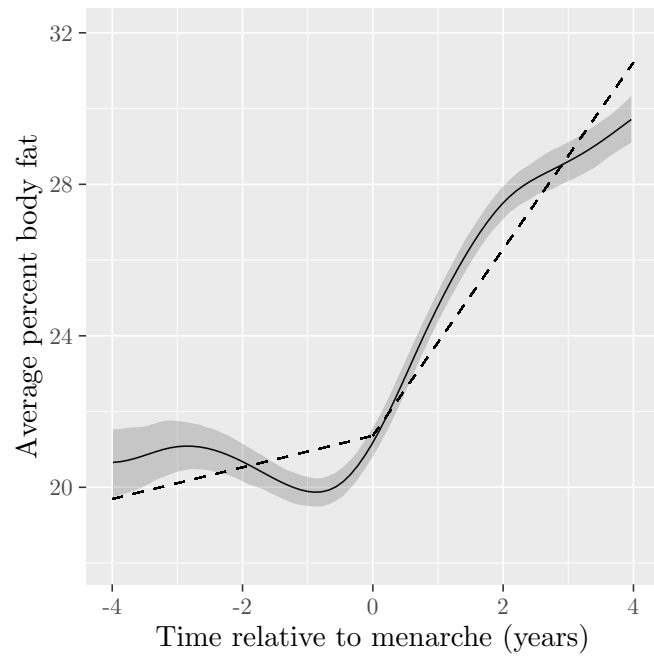
Figure 13: The population-averaged percent body fat over time, with 95% confidence intervals. The dashed line is the estimated population-averaged curve using the piecewise linear model from Fitzmaurice et al. (2011).
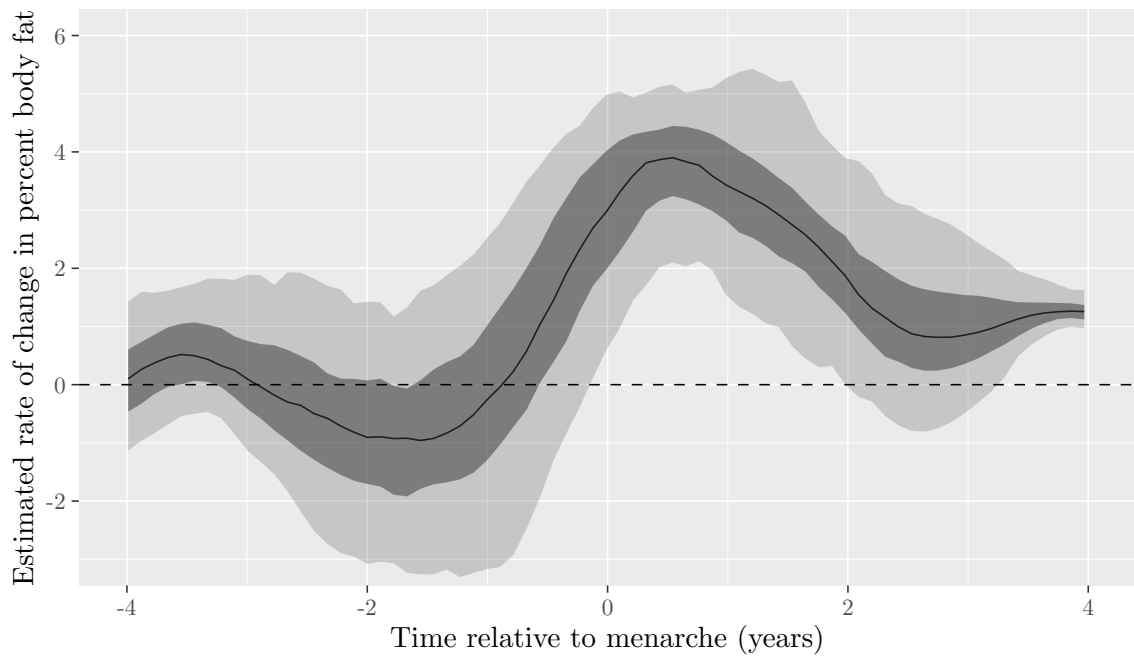
Figure 14: The distribution of the estimated rate of change in body fat across subjects. The solid line shows the median, the darker shaded region is 25th to 75th quantile, the lighter shaded region is 5th to 95th quantile.
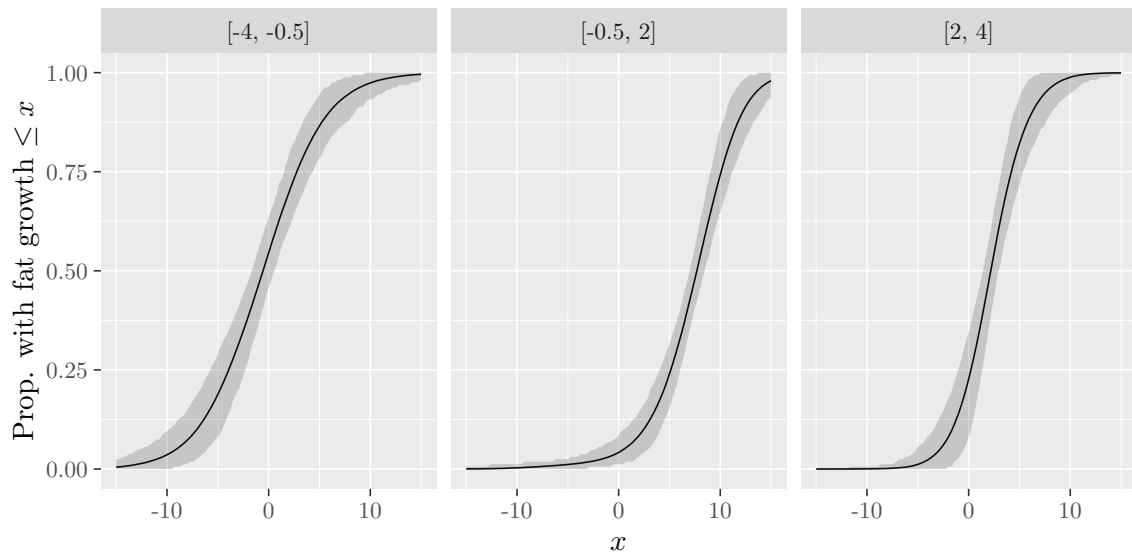
Figure 15: The distribution of change of body fat across subjects, during each of three time intervals (times are in years relative to menarche). The solid line is the estimated cumulative distribution function, and the shaded region gives a 95% confidence interval.

months after menarche. There is far more diversity among subjects before menarche, with three quarters of subjects estimated to have decreasing body fat around 2 years before menarche.

Figure 14 does not incorporate uncertainty in our estimates. We can show the uncertainty in subject-specific behaviour if we instead focus on what happens in a fixed time interval, and look at the distribution of change in body fat during that interval, across subjects. We split the time of the study into three time intervals: 4 years to 6 months before menarche (where the average body fat is approximately constant), 6 months before to 2 years after menarche (where there is a rapid growth in average body fat) and 2 to 4 years after menarche (where there is a slower growth in average body fat). We can consider the distribution of subject-specific changes in body fat over each of these intervals. Figure 15 shows an estimated cumulative distribution function for change of body fat in each time interval across subjects, with associated 95% confidence intervals. In each case, the cumulative distribution function at x is the proportion of subjects with fat growth in the interval $\leq x$.

In the time 4 years to 6 months before menarche, there is a lot of variation between subjects: we can be confident that some experience large increases in percent body fat while others experience large decreases (of at least 5 percentage points either way) in this time. The vast majority (91–99%) have increasing percent body fat 6 months before to 2 years after menarche, with most (69–85%) increasing by more than 5 percentage points in this interval. Most subjects continue to increase in body fat 2–4 years post menarche, though at a slower rate, with considerable uncertainty about the proportion of subjects who have decreasing body fat in this time (7–25%).

7 Discussion and Future Work

The setup of this paper was simple: modelling subject-specific response over time, without additional explanatory variables. Despite the simplicity of the setup, all existing methods considered failed in some of the cases we studied. Simple mixed-effects models make strong assumptions about the relationship between the subject-specific mean curves (e.g. that they vary only in intercept or slope). Functional principal components analysis allows for general relationships between the curves, but existing methods, such as PACE, do not work well in many of our simulated ex-

amples, particularly when the truth is a simple mixed-effects model. Our new penalised likelihood approach to functional principal components analysis offers much improved inference.

Future work will focus on extensions to make the methodology more widely applicable, by allowing for dependence on additional explanatory variables in the model, and generalising to different types of response distribution (for instance to allow for binary or count data).

This paper described models for longitudinal data, but all of these models may also be used for clustered data, with clusters i, and any continuous explanatory variable in place of time t. Mixed-effects models are also used for more complex clustered data, such as multiple nested levels of clustering. Another avenue for future work is to consider the extension of these models to allow for more complex clustering.

Code availability

The new penalised likelihood method is implemented in R package flexl, available at
`https://github.com/heogden/flexl`

Acknowledgements

References

Chong-Zhi Di, Ciprian M. Crainiceanu, Brian S. Caffo, and Naresh M. Punjabi. Multilevel functional principal component analysis. *The Annals of Applied Statistics*, 3(1):458 – 488, 2009. doi: 10.1214/08-AOAS206. URL `https://doi.org/10.1214/08-AOAS206`.

Garrett M Fitzmaurice, Nan M Laird, and James H Ware. *Applied Longitudinal Analysis*. John Wiley & Sons, second edition, 2011. doi: 10.1002/9781119513469.

J. Goldsmith, S. Greven, and C. Crainiceanu. Corrected confidence bands for functional data using principal components. *Biometrics*, 69(1):41–51, 2013. doi: https://doi.org/10.1111/j.1541-0420.2012.01808.x.

Jeff Goldsmith, Fabian Scheipl, Lei Huang, Julia Wrobel, Chongzhi Di, Jonathan Gellar, Jaroslaw Harezlak, Mathew W. McLean, Bruce Swihart, Luo Xiao, Ciprian Crainiceanu, and Philip T. Reiss. *refund: Regression with Functional Data*, 2023. URL `https://CRAN.R-project.org/package=refund`. R package version 0.1-30.

Sebastian P. Luque and Douglas Bates. *ALA: Data sets and examples for the book "Applied Longitudinal Analysis"*, 2012. URL `https://R-Forge.R-project.org/projects/applong/`. R package version 1.0/r53.

Eric J Pedersen, David L Miller, Gavin L Simpson, and Noam Ross. Hierarchical generalized additive models in ecology: an introduction with mgcv. *PeerJ*, 7:e6876, 2019.

Sarah M. Phillips, Linda G. Bandini, Dung V. Compton, Elena N. Naumova, and Aviva Must. A longitudinal comparison of body composition by total body water and bioelectrical impedance in adolescent girls. *The Journal of Nutrition*, 133(5):1419–1425, 2003. ISSN 0022-3166. doi: https://doi.org/10.1093/jn/133.5.1419. URL `https://www.sciencedirect.com/science/article/pii/S0022316622158736`.

Andrew Redd. *orthogonalsplinebasis: Orthogonal B-Spline Basis Functions*, 2022. URL `https://CRAN.R-project.org/package=orthogonalsplinebasis`. R package version 0.1.7.

Fabian Scheipl, Ana-Maria Staicu, and Sonja Greven. Functional additive mixed models. *Journal of Computational and Graphical Statistics*, 24(2):477–501, 2015. doi: 10.1080/10618600.2014.901914.

Stan Development Team. StanHeaders: Headers for the R interface to Stan, 2020. URL https://mc-stan.org/. R package version 2.21.0-6.

S. N. Wood. *Generalized Additive Models: An Introduction with R.* Chapman and Hall/CRC, second edition, 2017.

Fang Yao, Hans-Georg Müller, and Jane-Ling Wang. Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association*, 100(470):577–590, 2005. doi: 10.1198/016214504000001745.

Yidong Zhou, Satarupa Bhattacharjee, Cody Carroll, Yaqing Chen, Xiongtao Dai, Jianing Fan, Alvaro Gajardo, Pantelis Z. Hadjipantelis, Kyunghee Han, Hao Ji, Changbo Zhu, Hans-Georg Müller, and Jane-Ling Wang. *fdapace: Functional Data Analysis and Empirical Dynamics*, 2022. URL https://CRAN.R-project.org/package=fdapace. R package version 0.5.9.

A Proofs

Proof of Claim 1. By differentiating (10) twice, we have

$$\mu_i''(t) = f_0''(t) + \sum_{k=1}^{K} u_{ik} f_k''(t),$$

so

$$[\mu_i''(t)]^2 = (f_0''(t))^2 + 2 f_0''(t) \sum_{k=1}^{K} u_{ik} f_k''(t) + \sum_{k=1}^{K} \sum_{l=1}^{K} u_{ik} u_{il} f_k''(t) f_l''(t).$$

So

$$w(\mu_i) = w(f_0) + 2 \sum_{k=1}^{K} u_{ik} \int_{-\infty}^{\infty} f_0''(t) f_k''(t) dt + \sum_{k=1}^{K} \sum_{l=1}^{K} u_{ik} u_{il} \int_{-\infty}^{\infty} f_k''(t) f_l''(t) dt$$

and

$$w_E = E[w(\mu_i)]$$

$$= w(f_0) + 2 \sum_{k=1}^{K} E(u_{ik}) \int_{-\infty}^{\infty} f_0''(t) f_k''(t) dt + \sum_{k=1}^{K} \sum_{l=1}^{K} E(u_{ik} u_{il}) \int_{-\infty}^{\infty} f_k''(t) f_l''(t) dt$$

$$= w(f_0) + \sum_{k=1}^{K} E(u_{ik}^2) \int_{-\infty}^{\infty} (f_k''(t))^2 dt$$

since u_{ik} are independent, with mean zero

$$= w(f_0) + \sum_{k=1}^{K} w(f_k),$$

since $E(u_{ik}^2) = \text{Var}(u_{ik}) = 1$, as required. □

Proof of Claim 2. For any i and j, we have

$$\langle f_i, f_j \rangle = \langle \sum_{k=1}^{n_B} \beta_{ik} b_k, \sum_{l=1}^{n_B} \beta_{jl} b_l \rangle$$

$$= \sum_{k=1}^{n_B} \sum_{l=1}^{n_B} \beta_{ik} \beta_{jl} \langle b_k, b_l \rangle$$

$$= \sum_{l=1}^{n_B} \beta_{il} \beta_{jl} = \langle \beta_i, \beta_j \rangle$$

since $\langle b_k, b_l \rangle = 0$ if $k \neq l$ and $\langle b_l, b_l \rangle = 1$.

So $\langle f_i, f_j \rangle = 0$ if and only if $\beta_i, \beta_j \rangle = 0$, and the result follows. □

B   Tables from simulation studies

Table 1: Root Mean Integrated Squared Errors for a variety of estimation methods in the random intercept simulations.

σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PACE 95	PACE 99	PACE AIC	PACE BIC	PACE Oracle	PL	RI	RS
0.5	0.1	20	5	0.92	459.57	0.18	0.27	0.29	0.29	0.29	0.20	0.05	0.05	0.05
0.5	0.1	50	5	0.22	942.96	0.12	0.18	0.20	0.20	0.20	0.14	0.05	0.05	0.05
0.5	0.1	20	10	0.12	0.13	0.13	0.10	0.10	0.11	0.11	0.10	0.03	0.03	0.03
0.5	0.1	50	10	0.10	0.12	0.09	0.09	0.12	0.12	0.12	0.09	0.03	0.03	0.03
1	0.1	20	5	0.68	57.03	0.34	0.48	0.55	0.55	0.55	0.36	0.05	0.05	0.05
1	0.1	50	5	0.49	967.64	0.22	0.44	0.47	0.47	0.47	0.33	0.05	0.05	0.05
1	0.1	20	10	0.22	0.24	0.25	0.20	0.22	0.22	0.22	0.20	0.03	0.03	0.03
1	0.1	50	10	0.17	0.19	0.18	0.17	0.21	0.21	0.21	0.16	0.03	0.03	0.03
0.5	0.5	20	5	0.52	2.71	0.29	0.41	0.42	0.42	0.42	0.31	0.23	0.22	0.23
0.5	0.5	50	5	174.17	0.45	0.24	0.23	0.23	0.23	0.23	0.23	0.22	0.21	0.22
0.5	0.5	20	10	0.23	0.24	0.21	0.19	0.19	0.19	0.19	0.19	0.16	0.16	0.16
0.5	0.5	50	10	0.21	0.21	0.18	0.17	0.17	0.17	0.17	0.17	0.16	0.15	0.15
1	0.5	20	5	1.31	687.29	0.41	0.95	1.01	1.01	1.01	0.72	0.25	0.23	0.25
1	0.5	50	5	2.21	5355.53	0.31	0.28	0.29	0.28	0.28	0.28	0.23	0.23	0.23
1	0.5	20	10	0.33	0.38	0.30	0.29	0.30	0.30	0.30	0.26	0.17	0.16	0.17
1	0.5	50	10	0.28	0.34	0.24	0.50	0.60	0.60	0.60	0.34	0.16	0.16	0.16

Table 2: Root Median Integrated Squared Errors for a variety of estimation methods in the random intercept simulations.

σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PACE 95	PACE 99	PACE AIC	PACE BIC	PACE Oracle	PL	RI	RS
0.5	0.1	20	5	0.13	0.14	0.11	0.09	0.09	0.09	0.09	0.09	0.04	0.03	0.04
0.5	0.1	50	5	0.09	0.11	0.08	0.07	0.07	0.07	0.07	0.07	0.03	0.03	0.03
0.5	0.1	20	10	0.08	0.09	0.09	0.07	0.07	0.07	0.07	0.07	0.03	0.02	0.03
0.5	0.1	50	10	0.07	0.08	0.06	0.05	0.06	0.06	0.06	0.05	0.02	0.02	0.02
1	0.1	20	5	0.21	0.22	0.20	0.14	0.14	0.14	0.14	0.15	0.04	0.03	0.04
1	0.1	50	5	0.15	0.18	0.14	0.12	0.12	0.12	0.12	0.12	0.03	0.03	0.03
1	0.1	20	10	0.14	0.14	0.16	0.13	0.12	0.12	0.12	0.13	0.02	0.02	0.03
1	0.1	50	10	0.11	0.12	0.12	0.09	0.09	0.09	0.09	0.09	0.02	0.02	0.02
0.5	0.5	20	5	0.26	0.26	0.23	0.20	0.20	0.20	0.20	0.19	0.17	0.15	0.17
0.5	0.5	50	5	0.19	0.19	0.17	0.17	0.17	0.17	0.17	0.17	0.15	0.14	0.15
0.5	0.5	20	10	0.19	0.18	0.16	0.14	0.14	0.14	0.14	0.14	0.12	0.11	0.12
0.5	0.5	50	10	0.17	0.17	0.14	0.13	0.13	0.13	0.13	0.13	0.11	0.10	0.11
1	0.5	20	5	0.35	0.37	0.29	0.25	0.25	0.25	0.25	0.25	0.18	0.16	0.18
1	0.5	50	5	0.30	0.31	0.24	0.21	0.21	0.21	0.21	0.21	0.17	0.16	0.17
1	0.5	20	10	0.26	0.27	0.23	0.19	0.19	0.19	0.19	0.19	0.12	0.11	0.12
1	0.5	50	10	0.22	0.24	0.19	0.16	0.17	0.16	0.16	0.16	0.11	0.11	0.11

Table 3: Coverage of 95% confidence intervals for a variety of methods in the random intercept simulations.

σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PL
0.5	0.1	20	5	0.26	0.26	0.39	0.94
0.5	0.1	50	5	0.29	0.27	0.45	0.95
0.5	0.1	20	10	0.27	0.25	0.34	0.95
0.5	0.1	50	10	0.31	0.28	0.48	0.95
1	0.1	20	5	0.21	0.20	0.31	0.93
1	0.1	50	5	0.20	0.17	0.34	0.94
1	0.1	20	10	0.20	0.19	0.28	0.95
1	0.1	50	10	0.23	0.16	0.35	0.95
0.5	0.5	20	5	0.73	0.74	0.73	0.94
0.5	0.5	50	5	0.87	0.89	0.86	0.95
0.5	0.5	20	10	0.85	0.87	0.82	0.95
0.5	0.5	50	10	0.91	0.92	0.88	0.96
1	0.5	20	5	0.46	0.46	0.59	0.94
1	0.5	50	5	0.61	0.61	0.73	0.95
1	0.5	20	10	0.60	0.61	0.69	0.95
1	0.5	50	10	0.74	0.73	0.79	0.96

Table 4: Root Mean Integrated Squared Errors for a variety of estimation methods in the random slope simulations.

ρ	σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PACE 95	PACE 99	PACE AIC	PACE BIC	PACE Oracle	PL	RI	RS
0	0.5	0.1	20	5	7.58	4914.53	0.27	0.37	0.39	0.39	0.39	0.34	0.08	0.16	0.08
0	0.5	0.1	50	5	0.66	192.77	0.20	0.34	0.36	0.36	0.36	0.33	0.08	0.17	0.08
0	0.5	0.1	20	10	0.17	0.20	0.17	0.18	0.19	0.19	0.19	0.16	0.05	0.15	0.05
0	0.5	0.1	50	10	0.15	0.16	0.15	0.17	0.18	0.18	0.18	0.16	0.05	0.15	0.05
0	1	0.1	20	5	0.99	4212.21	0.40	0.56	0.58	0.58	0.58	0.53	0.08	0.16	0.08
0	1	0.1	50	5	0.69	401.69	0.31	0.33	0.38	0.39	0.39	0.37	0.08	0.17	0.08
0	1	0.1	20	10	0.27	0.29	0.27	0.29	0.34	0.34	0.34	0.31	0.05	0.15	0.05
0	1	0.1	50	10	0.22	0.23	0.22	0.24	0.25	0.26	0.26	0.23	0.05	0.15	0.05
0	0.5	0.5	20	5	5.98	9609.34	0.39	0.34	0.40	0.40	0.40	0.34	0.27	0.27	0.26
0	0.5	0.5	50	5	0.31	0.32	0.30	0.27	0.27	0.27	0.27	0.27	0.26	0.27	0.25
0	0.5	0.5	20	10	0.28	0.29	0.27	0.23	0.23	0.23	0.23	0.23	0.20	0.22	0.20
0	0.5	0.5	50	10	0.24	0.25	0.24	0.21	0.21	0.21	0.21	0.21	0.19	0.21	0.19
0	1	0.5	20	5	2.03	543.89	0.49	0.95	1.45	1.45	1.45	1.08	0.28	0.27	0.28
0	1	0.5	50	5	4.83	1358.93	0.47	0.32	0.32	0.32	0.32	0.32	0.27	0.28	0.27
0	1	0.5	20	10	0.35	0.42	0.34	0.33	0.36	0.36	0.36	0.33	0.20	0.22	0.20
0	1	0.5	50	10	0.31	0.37	0.30	0.42	0.46	0.46	0.46	0.42	0.20	0.22	0.20
0.5	0.5	0.1	20	5	3.62	44.83	0.30	0.34	0.38	0.38	0.38	0.35	0.07	0.16	0.07
0.5	0.5	0.1	50	5	0.50	19019.98	0.21	0.30	0.35	0.35	0.35	0.30	0.07	0.17	0.07
0.5	0.5	0.1	20	10	0.19	0.19	0.18	0.18	0.19	0.19	0.19	0.18	0.05	0.15	0.05
0.5	0.5	0.1	50	10	0.15	0.16	0.15	0.17	0.20	0.20	0.20	0.17	0.05	0.15	0.05
0.5	1	0.1	20	5	3.96	1057.61	0.48	0.61	0.63	0.63	0.64	0.59	0.08	0.16	0.07
0.5	1	0.1	50	5	0.47	356.50	0.31	0.41	0.45	0.45	0.45	0.43	0.08	0.17	0.07
0.5	1	0.1	20	10	0.28	0.30	0.28	0.28	0.35	0.35	0.35	0.30	0.05	0.15	0.05
0.5	1	0.1	50	10	0.21	0.24	0.22	0.25	0.27	0.27	0.28	0.24	0.05	0.15	0.05
0.5	0.5	0.5	20	5	1.81	861.11	0.47	0.53	0.57	0.57	0.57	0.49	0.26	0.27	0.26
0.5	0.5	0.5	50	5	0.37	0.74	0.32	0.27	0.27	0.27	0.27	0.27	0.25	0.27	0.25
0.5	0.5	0.5	20	10	0.29	0.30	0.27	0.23	0.23	0.23	0.23	0.23	0.19	0.22	0.19
0.5	0.5	0.5	50	10	0.25	0.25	0.24	0.21	0.21	0.20	0.20	0.21	0.18	0.22	0.18
0.5	1	0.5	20	5	1.73	205.65	0.59	0.77	1.17	1.17	1.17	1.04	0.27	0.27	0.27
0.5	1	0.5	50	5	3.64	1338.11	0.51	0.91	0.92	0.92	0.92	0.91	0.26	0.28	0.26
0.5	1	0.5	20	10	0.36	0.43	0.36	0.34	0.38	0.38	0.38	0.34	0.20	0.22	0.19
0.5	1	0.5	50	10	0.32	0.38	0.31	0.45	0.53	0.53	0.53	0.45	0.19	0.22	0.19

Table 5: Root Median Integrated Squared Errors for a variety of estimation methods in the random slope simulations.

ρ	σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PACE 95	PACE 99	PACE AIC	PACE BIC	PACE Oracle	PL	RI	RS
0	0.5	0.1	20	5	0.17	0.20	0.17	0.12	0.11	0.12	0.12	0.11	0.06	0.11	0.06
0	0.5	0.1	50	5	0.15	0.16	0.13	0.12	0.11	0.11	0.11	0.11	0.06	0.12	0.06
0	0.5	0.1	20	10	0.12	0.12	0.13	0.11	0.10	0.10	0.10	0.10	0.04	0.11	0.04
0	0.5	0.1	50	10	0.11	0.11	0.11	0.10	0.08	0.08	0.08	0.08	0.04	0.11	0.04
0	1	0.1	20	5	0.25	0.30	0.24	0.18	0.16	0.17	0.17	0.17	0.06	0.11	0.06
0	1	0.1	50	5	0.20	0.22	0.20	0.17	0.16	0.16	0.16	0.16	0.06	0.12	0.06
0	1	0.1	20	10	0.18	0.18	0.18	0.16	0.15	0.16	0.16	0.15	0.04	0.11	0.04
0	1	0.1	50	10	0.16	0.16	0.16	0.14	0.13	0.14	0.14	0.13	0.04	0.11	0.04
0	0.5	0.5	20	5	0.27	0.26	0.26	0.23	0.23	0.23	0.23	0.23	0.22	0.21	0.21
0	0.5	0.5	50	5	0.25	0.25	0.25	0.22	0.22	0.22	0.22	0.22	0.20	0.21	0.20
0	0.5	0.5	20	10	0.23	0.23	0.23	0.19	0.18	0.19	0.19	0.18	0.16	0.18	0.16
0	0.5	0.5	50	10	0.21	0.21	0.20	0.18	0.18	0.18	0.18	0.18	0.16	0.17	0.15
0	1	0.5	20	5	0.36	0.37	0.35	0.27	0.27	0.27	0.27	0.27	0.22	0.22	0.22
0	1	0.5	50	5	0.32	0.33	0.31	0.26	0.25	0.25	0.25	0.25	0.21	0.22	0.21
0	1	0.5	20	10	0.29	0.30	0.29	0.22	0.22	0.22	0.22	0.22	0.16	0.18	0.16
0	1	0.5	50	10	0.26	0.27	0.25	0.22	0.21	0.21	0.21	0.21	0.16	0.18	0.16
0.5	0.5	0.1	20	5	0.17	0.18	0.17	0.12	0.12	0.12	0.12	0.12	0.06	0.11	0.06
0.5	0.5	0.1	50	5	0.14	0.16	0.14	0.12	0.11	0.11	0.11	0.11	0.06	0.12	0.06
0.5	0.5	0.1	20	10	0.13	0.13	0.13	0.11	0.10	0.11	0.11	0.10	0.04	0.11	0.04
0.5	0.5	0.1	50	10	0.11	0.11	0.11	0.10	0.09	0.10	0.10	0.09	0.04	0.11	0.04
0.5	1	0.1	20	5	0.27	0.30	0.24	0.18	0.18	0.18	0.18	0.18	0.06	0.11	0.06
0.5	1	0.1	50	5	0.19	0.22	0.19	0.17	0.16	0.17	0.17	0.17	0.06	0.11	0.06
0.5	1	0.1	20	10	0.19	0.19	0.19	0.16	0.15	0.16	0.16	0.15	0.04	0.11	0.04
0.5	1	0.1	50	10	0.15	0.16	0.16	0.14	0.13	0.14	0.14	0.13	0.04	0.10	0.04
0.5	0.5	0.5	20	5	0.28	0.28	0.27	0.22	0.22	0.22	0.22	0.22	0.20	0.22	0.20
0.5	0.5	0.5	50	5	0.26	0.26	0.25	0.21	0.21	0.21	0.21	0.21	0.19	0.21	0.19
0.5	0.5	0.5	20	10	0.23	0.24	0.23	0.18	0.19	0.18	0.18	0.19	0.15	0.17	0.15
0.5	0.5	0.5	50	10	0.21	0.21	0.20	0.16	0.16	0.16	0.16	0.17	0.15	0.17	0.14
0.5	1	0.5	20	5	0.39	0.41	0.37	0.27	0.27	0.27	0.27	0.27	0.21	0.22	0.21
0.5	1	0.5	50	5	0.33	0.33	0.33	0.25	0.25	0.25	0.25	0.25	0.20	0.22	0.20
0.5	1	0.5	20	10	0.30	0.32	0.30	0.22	0.22	0.22	0.22	0.23	0.16	0.18	0.15
0.5	1	0.5	50	10	0.25	0.28	0.25	0.21	0.21	0.21	0.21	0.21	0.15	0.18	0.15

Table 6: Coverage of 95% confidence intervals for a variety of methods in the random slope simulations.

ρ	σ_u	σ	d	n_i	**Di 95**	**Di 99**	**Di Oracle**	**PL**
0	0.5	0.1	20	5	0.25	0.26	0.29	0.94
0	0.5	0.1	50	5	0.30	0.28	0.33	0.94
0	0.5	0.1	20	10	0.24	0.23	0.27	0.95
0	0.5	0.1	50	10	0.26	0.26	0.29	0.95
0	1	0.1	20	5	0.19	0.19	0.22	0.94
0	1	0.1	50	5	0.24	0.24	0.26	0.94
0	1	0.1	20	10	0.24	0.24	0.26	0.95
0	1	0.1	50	10	0.20	0.20	0.22	0.95
0	0.5	0.5	20	5	0.76	0.78	0.76	0.93
0	0.5	0.5	50	5	0.86	0.87	0.86	0.94
0	0.5	0.5	20	10	0.80	0.83	0.80	0.93
0	0.5	0.5	50	10	0.87	0.90	0.86	0.94
0	1	0.5	20	5	0.49	0.51	0.54	0.93
0	1	0.5	50	5	0.61	0.63	0.66	0.94
0	1	0.5	20	10	0.63	0.64	0.66	0.93
0	1	0.5	50	10	0.68	0.68	0.71	0.94
0.5	0.5	0.1	20	5	0.26	0.26	0.28	0.93
0.5	0.5	0.1	50	5	0.27	0.26	0.30	0.94
0.5	0.5	0.1	20	10	0.23	0.24	0.25	0.94
0.5	0.5	0.1	50	10	0.22	0.22	0.24	0.95
0.5	1	0.1	20	5	0.22	0.22	0.24	0.94
0.5	1	0.1	50	5	0.24	0.23	0.26	0.94
0.5	1	0.1	20	10	0.23	0.23	0.25	0.95
0.5	1	0.1	50	10	0.21	0.19	0.22	0.95
0.5	0.5	0.5	20	5	0.69	0.71	0.72	0.93
0.5	0.5	0.5	50	5	0.82	0.83	0.83	0.94
0.5	0.5	0.5	20	10	0.77	0.79	0.77	0.93
0.5	0.5	0.5	50	10	0.86	0.89	0.86	0.95
0.5	1	0.5	20	5	0.43	0.45	0.47	0.93
0.5	1	0.5	50	5	0.56	0.57	0.61	0.94
0.5	1	0.5	20	10	0.58	0.58	0.61	0.93
0.5	1	0.5	50	10	0.64	0.62	0.67	0.94

Table 7: Root Mean Integrated Squared Errors for a variety of estimation methods in the one-dimensional variation simulations.

σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PACE 95	PACE 99	PACE AIC	PACE BIC	PACE Oracle	PL	RI	RS
0.5	0.1	20	5	0.38	256.90	0.23	0.35	0.42	0.42	0.42	0.31	0.08	0.40	0.42
0.5	0.1	50	5	0.22	0.85	0.16	0.34	0.40	0.40	0.40	0.28	0.06	0.41	0.42
0.5	0.1	20	10	0.15	0.15	0.15	0.17	0.19	0.19	0.19	0.16	0.05	0.38	0.39
0.5	0.1	50	10	0.11	0.12	0.11	0.19	0.21	0.21	0.21	0.15	0.04	0.39	0.39
1	0.1	20	5	2.63	247.51	0.45	0.53	0.62	0.62	0.62	0.48	0.08	0.65	0.70
1	0.1	50	5	0.40	10.10	0.30	0.56	0.66	0.66	0.66	0.45	0.07	0.67	0.69
1	0.1	20	10	0.27	0.26	0.29	0.29	0.32	0.33	0.33	0.27	0.05	0.63	0.65
1	0.1	50	10	0.20	0.20	0.21	0.28	0.32	0.32	0.32	0.24	0.04	0.63	0.64
0.5	0.5	20	5	1.12	123.34	0.35	0.65	0.75	0.75	0.75	0.53	0.30	0.45	0.47
0.5	0.5	50	5	2.20	309.59	0.30	0.30	0.30	0.30	0.30	0.29	0.27	0.45	0.46
0.5	0.5	20	10	0.27	0.28	0.24	0.24	0.24	0.24	0.24	0.23	0.21	0.41	0.42
0.5	0.5	50	10	0.23	0.33	0.20	0.47	0.58	0.58	0.58	0.33	0.19	0.41	0.41
1	0.5	20	5	1.37	1766.13	0.50	2.59	3.15	3.15	3.15	2.18	0.34	0.69	0.72
1	0.5	50	5	1.89	1833.27	0.40	1.90	2.32	2.32	2.32	1.36	0.29	0.70	0.71
1	0.5	20	10	0.37	0.40	0.34	0.56	0.69	0.69	0.69	0.47	0.22	0.65	0.66
1	0.5	50	10	0.31	0.36	0.27	0.98	1.11	1.11	1.11	0.69	0.20	0.65	0.65

Table 8: Root Median Integrated Squared Errors for a variety of estimation methods in the one-dimensional variation simulations.

σ_u	σ	d	n_i	Di 95	Di 99	Di Oracle	PACE 95	PACE 99	PACE AIC	PACE BIC	PACE Oracle	PL	RI	RS
0.5	0.1	20	5	0.15	0.17	0.15	0.16	0.17	0.17	0.17	0.15	0.06	0.29	0.30
0.5	0.1	50	5	0.12	0.13	0.12	0.13	0.13	0.13	0.13	0.13	0.05	0.29	0.30
0.5	0.1	20	10	0.10	0.10	0.11	0.10	0.10	0.10	0.10	0.10	0.04	0.27	0.27
0.5	0.1	50	10	0.08	0.08	0.08	0.08	0.09	0.09	0.09	0.08	0.03	0.27	0.28
1	0.1	20	5	0.25	0.27	0.26	0.24	0.25	0.25	0.25	0.24	0.06	0.47	0.47
1	0.1	50	5	0.18	0.20	0.19	0.20	0.22	0.22	0.22	0.20	0.05	0.46	0.47
1	0.1	20	10	0.17	0.16	0.18	0.17	0.16	0.17	0.17	0.16	0.04	0.41	0.43
1	0.1	50	10	0.13	0.12	0.14	0.14	0.14	0.14	0.14	0.14	0.04	0.43	0.43
0.5	0.5	20	5	0.29	0.29	0.27	0.27	0.27	0.26	0.26	0.26	0.23	0.36	0.37
0.5	0.5	50	5	0.24	0.24	0.22	0.22	0.22	0.22	0.22	0.21	0.19	0.36	0.36
0.5	0.5	20	10	0.20	0.20	0.18	0.19	0.19	0.19	0.18	0.18	0.16	0.31	0.32
0.5	0.5	50	10	0.17	0.18	0.16	0.16	0.17	0.16	0.16	0.15	0.14	0.32	0.32
1	0.5	20	5	0.41	0.44	0.36	0.38	0.38	0.38	0.38	0.35	0.27	0.50	0.51
1	0.5	50	5	0.33	0.36	0.29	0.31	0.31	0.31	0.31	0.30	0.22	0.50	0.51
1	0.5	20	10	0.26	0.28	0.24	0.23	0.24	0.23	0.23	0.23	0.18	0.45	0.45
1	0.5	50	10	0.23	0.25	0.21	0.22	0.23	0.22	0.22	0.20	0.15	0.46	0.46

Table 9: Coverage of 95% confidence intervals for a variety of methods in the one-dimensional variation simulations.

σ_u	σ	d	n_i	**Di 95**	**Di 99**	**Di Oracle**	**PL**
0.5	0.1	20	5	0.39	0.41	0.47	0.95
0.5	0.1	50	5	0.46	0.48	0.55	0.96
0.5	0.1	20	10	0.41	0.43	0.42	0.96
0.5	0.1	50	10	0.37	0.37	0.45	0.96
1	0.1	20	5	0.38	0.40	0.46	0.94
1	0.1	50	5	0.46	0.48	0.54	0.95
1	0.1	20	10	0.38	0.40	0.40	0.94
1	0.1	50	10	0.34	0.35	0.40	0.93
0.5	0.5	20	5	0.65	0.67	0.64	0.94
0.5	0.5	50	5	0.76	0.79	0.75	0.96
0.5	0.5	20	10	0.76	0.81	0.71	0.95
0.5	0.5	50	10	0.82	0.86	0.79	0.97
1	0.5	20	5	0.47	0.48	0.54	0.94
1	0.5	50	5	0.57	0.58	0.63	0.96
1	0.5	20	10	0.53	0.56	0.56	0.96
1	0.5	50	10	0.59	0.61	0.63	0.97