Functional Mixed-effect Model in Quantitative Genetics: Research Project Outline

Genetic Framework

- Quantitative genetics studys the quantitative characteristics that are characterised by the presence of continuous type of variation in a population.
- Function-valued (FV) traits, such as growth trajectories are phenotypes of living organisms whose value can be described by a function of some continuous index. This functional nature suggests the potential of using modern functional data analysis in quantitative genetics.
- A key concept here is that the variation between individuals can be decomposed into genetic composition and environmental effects.

$$Var(y) = V_G + V_E$$

  In particular, the genetic variation is estimated by comparing phenotypical variation in related individuals where the genetic relationship is modelled by the additive relationship matrix $\boldsymbol{A}$. It is a positive definite matrix with each term A_{ij} representing how closely related between individuals i and j. An example of $\boldsymbol{A}$:

$$\begin{pmatrix} 1 & 0.5 & 0.125 & 0 & \dots \\ 0 & 1 & 0.25 & 0 & \dots \\ 0 & 0.5 & 1 & 0.25 & \dots \\ \vdots & & & & \end{pmatrix}$$

- In a functional setting, the change in mean phenotype in one generration of selection, $\Delta\bar{y}(t)$, can be modelled by

$$\Delta\bar{y}(t) = \int_{\mathbb{R}} G(s,t)\beta(s)\,ds$$

  where $G(s,t)$ is the additive genetic covariance function and $\beta(s)$ is the selection gradient function. The selection gradient in the direction of the first eigenfunction of the additive genetic covariance would result in the maximum response to selection. Therefore, our main goal is to estimate the additive genetic covariance function.

Data Description

In this project, we will be working on the $Triboleum\ Castaneum$ dataset, $\mathrm{TRFUN25PUP4}$. All $Tribolium$ individuals were derived from a stock population of the cSM++ strain, with approximately 300 individuals initially divided into 17 populations of about 20 individuals each and allowed to breed freely. Offspring were collected from these populations as they pupated, their sex was determined and recorded, and they were isolated into separate vials and used to create a stock of isolated, virgin adults.

A half-sib/full-sib breeding design was used to facilitate quantitative genetic analysis. From the stock of isolated virgin adults, one randomly chosen male was mated with five randomly chosen females, none of which were his siblings. This was repeated 30 times, using a total of 30 males and 150 females, thereby producing 30 half-sib families and 150 full-sib families. The sire, dam and subjects' ids in the the dataset will be used to build the pedigree and extract the additive genetic relationship matrix $\boldsymbol{A}$.

The trait reported in this dataset is the body mass measured for each lava at different ages during the larval period and mass at pupation was included as the final mass measure for each growth curve. It contains 873 individuals and 6860 measurements, with an average of approximately 8 measurements per individual. Sampling points are not taken at fixed times as they vary in number and location, the range of days measured is 1-25 days.

Functional Mixed-Effect Model

We are interested in fitting a functional mixed-effect model (FMEM) represented by basis functions:

$$\underbrace{Y_{ij}}_{\text{trait}} = \underbrace{\mu(t_{ij})}_{\text{fixed effect: population mean}} + \underbrace{\sum_{k=1}^{K}\alpha_{ik}\phi(t_{ij})}_{\text{genetic random effect}} + \underbrace{\sum_{k=1}^{K}\gamma_{ik}\phi(t_{ij})}_{\text{environmental random effect}} + \underbrace{\epsilon_{ij}}_{\text{measurement error}}$$

In matrix form:

$$\begin{aligned}\boldsymbol{Y} &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z^G}\boldsymbol{\alpha} + \boldsymbol{Z^E}\boldsymbol{\gamma} + \boldsymbol{\epsilon} \\ &= \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u} + \boldsymbol{\epsilon}\end{aligned}$$

where $\boldsymbol{X}$ is the predictor matrix; $\boldsymbol{\beta}$ is the vector fixed-effect coefficients; $\boldsymbol{Z} = [\boldsymbol{Z^G}, \boldsymbol{Z^E}]$ is the design matrix for genetic and environmental random effects; $\boldsymbol{u} = [\boldsymbol{\alpha}, \boldsymbol{\gamma}]^{\boldsymbol{T}}$ with $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ are vectors for genetic and environmental random effect respectively; $\boldsymbol{\epsilon}$ is the meansurement error vector.

This is the random regression (RR) model reported in Meyer(1998). The random terms are assumed to have the following distributions:

$$\begin{aligned}\boldsymbol{\alpha} &\sim N(\boldsymbol{0}, \boldsymbol{A} \otimes \boldsymbol{C^G}) \\ \boldsymbol{\gamma} &\sim N(\boldsymbol{0}, \boldsymbol{I} \otimes \boldsymbol{C^E}) \\ \boldsymbol{\epsilon} &\sim N(\boldsymbol{0}, \sigma^2_{res} \otimes \boldsymbol{I})\end{aligned}$$

The covariance is

$$Cov(\boldsymbol{Y}_{ij}, \boldsymbol{Y}_{ij'}) = \sum_{k=1}^{K}\sum_{l=1}^{K} \phi_k(t_{ij})\phi_l(t_{ij'})\, Cov(\alpha_{ik}, \alpha_{il}) + \sum_{k=1}^{K}\sum_{l=1}^{K} \phi_k(t_{ij})\phi_l(t_{ij'})\, Cov(\gamma_{ik}, \gamma_{il}) + Cov(\epsilon_{ij}, \epsilon_{ij'})$$

Therefore, we can estimate the genetic covariance function provided that we know $Cov(\alpha_{ik}, \alpha_{il})$.

Model Reparameterisation

The package $\mathbf{lme4}$ provides functions to fit and estimate the parameters in linear mixed-effect models, generalised linear mixed-effect models and nonlinear mixed-effect models. However, this package cannot fit data with correlated subjects. A desired model should have the random effect matrix in the form of a block diagonal. In the genetic framework, the covariance matrix of $\boldsymbol{\alpha}$ is $\boldsymbol{A} \otimes \boldsymbol{C^G}$, which is clearly not block diagonal. Therefore, we need to reparameterise our model in order to use this package.

First, we take the Cholesky decomposition of $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L^T}$. Then define $\boldsymbol{M^{-1}} = \boldsymbol{L^{-1}} \otimes I_K$, assuming $dim(\boldsymbol{\alpha}) = K$. Next we take the new genetic random-effect matrix to be:

$$\tilde{\boldsymbol{Z^G}} = \boldsymbol{Z^G}\boldsymbol{M}$$

and the new genetic random-effect vector:

$$\tilde{\boldsymbol{\alpha}} = \boldsymbol{M}^{-1}\boldsymbol{\alpha}$$

We can check the covariance structure of $\tilde{\boldsymbol{\alpha}}$:

$$\begin{aligned}Cov(\tilde{\boldsymbol{\alpha}}) &= \boldsymbol{M}^{-1}(\boldsymbol{A} \otimes \boldsymbol{C^G})(\boldsymbol{M}^{-1})^{\boldsymbol{T}} \\ &= (\boldsymbol{L}^{-1} \otimes \boldsymbol{I_K})(\boldsymbol{L}\boldsymbol{L^T} \otimes \boldsymbol{C^G})(\boldsymbol{L}^{-1} \otimes \boldsymbol{I_K})^{\boldsymbol{T}} \\ &= (\boldsymbol{L^T} \otimes \boldsymbol{C^G})(\boldsymbol{L}^{-1} \otimes \boldsymbol{I_K})^{\boldsymbol{T}} \\ &= \boldsymbol{I} \otimes \boldsymbol{C^G}\end{aligned}$$

which is block diagnonal.

Now let us add the environmental effect and re-write our model in matrix form:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}^{*}\boldsymbol{u}^{*} + \boldsymbol{\epsilon}$$

where $\boldsymbol{Z}^{*} = [\boldsymbol{Z^G}\boldsymbol{M}, \boldsymbol{Z^E}]$ and $\boldsymbol{u}^{*} = [\boldsymbol{M}^{-1}\boldsymbol{\alpha}, \boldsymbol{\gamma}]^{\boldsymbol{T}}$. We are ready to fit the genetic functional mixed-effect model.

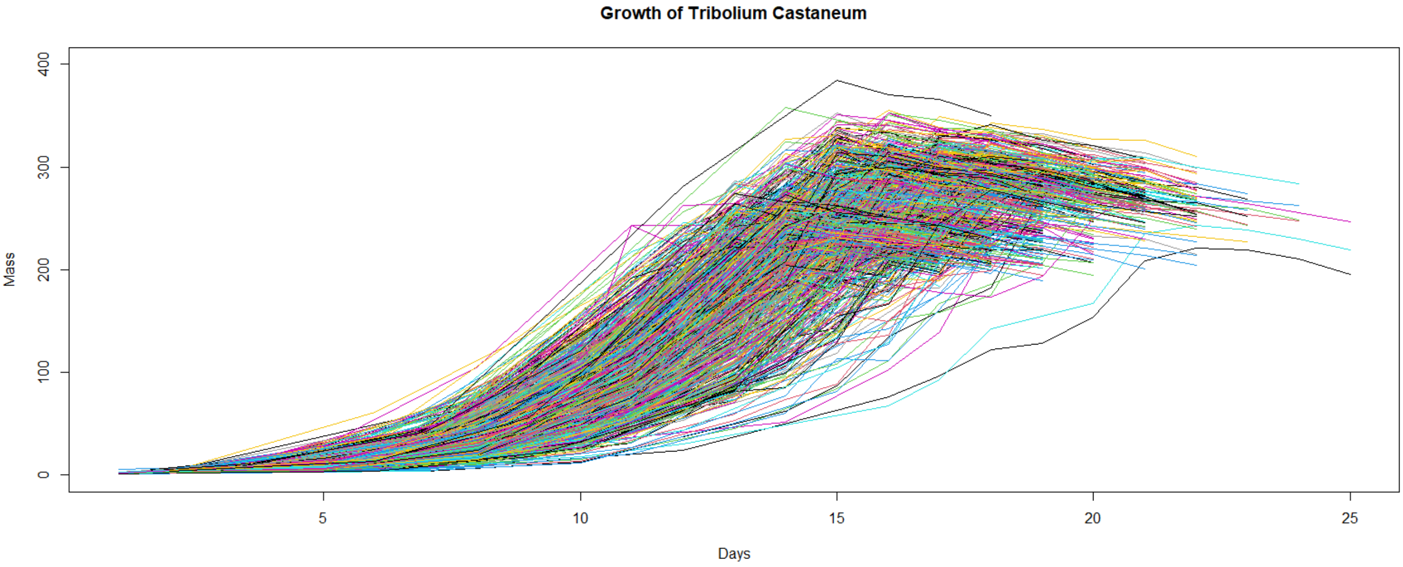
Model-Fitting Procedure

1. Data Smoothing

We use a cubic smoothing spline `smooth.spline` as the basis function to transform discrete data to functions. General cross-validation is used to choose the smoothing parameter λ.

2. Curve Alignment

The plot the growth curve of the raw data shows there are both phase and time variations in the data, and that confounding these two types of variations leads to many problems.



This issue, known as curve misalignment, can arise because that the rigid metric of physical time may not be directly relevant to the internal dynamics of many biological systems. Our main emphasis at this stage is on registration of the data by transformations of the argument t. We first standardise the timescale so that each growth curve is measured from 0 to 1, as proportion of the pupation time. Then we are looking for warping functions $h_i(t)$ such that the aligned curves $y(h_i(t))$ only presents amplitude variations. Here we consider the registration base on the Fisher-Rao metric proposed in Srivastava et al. (2011), implemented in the R package `fdasrvf` .

3. Functional Principal Component Analysis

A key approach in functional data analysis (FDA) is the functional principal component analysis (FPCA). This is partial because FPCA facilitates the conversion of infinite-dimensional functional data to a finite-dimension vector of random scores. The Mercer's Theorem implies that the spectral decomposition of the covariance function $v(s,t)$ leads to

$$v(s,t)=\sum_{j=1}^{\infty}\nu_j\xi_j(s)\xi_j(t)$$

where ν_j are the eigenvalues in descending order and ξ_j are the corresponding eigenfunctions.

By Karhunen-Lo$\grave{e}$ve (KL) expansion:

$$y_i(t)=\mu(t)+\sum_{j=1}^{\infty}S_{ij}\xi_j(t)$$

where $S_{ij}=\int(x_i(t)-\mu(t))\xi_j(t)\ dt$ are the FPC scores.

In practice, using K components, the FPC expansion can explain most variation in y, and the eigenfunctions $\xi(t)$ will be used as the basis functions $\phi(t)$ when fitting the mixed-effect model.

We estimate the covariance function of the aligned data on a fine grid of time points and then reducing the problem to the corresponding matrix spectral decomposition. This is done using `prcomp` in R.

4. Pedigree and Genetic Relationship Matrix

The additive genetic relationship matrix A encodes the degree of genetic relatedness among individuals based on their pedigree relationships. We have already stated the necessity of computing A in our model fitting procedure, and this matrix is computed by using the function `getA` from the R package `pedigreemm` . Once we know A, we compute its Cholesky factor and reparameterise our model as explained in the previous section.

5. Mixed-Effect Model Formula

Fit Fixed Effect

We consider two methods to fit the fixed-effect. The first way is to include a fixed regression of the same form as the random regression. Specifically, we use the same basis functions to represent $\mu(t)$ in our model. The second one is to estimate the population mean $\mu(t)$ based on working independent assumption:

$$y_i(t)=\mu(t)+\epsilon_i(t)$$

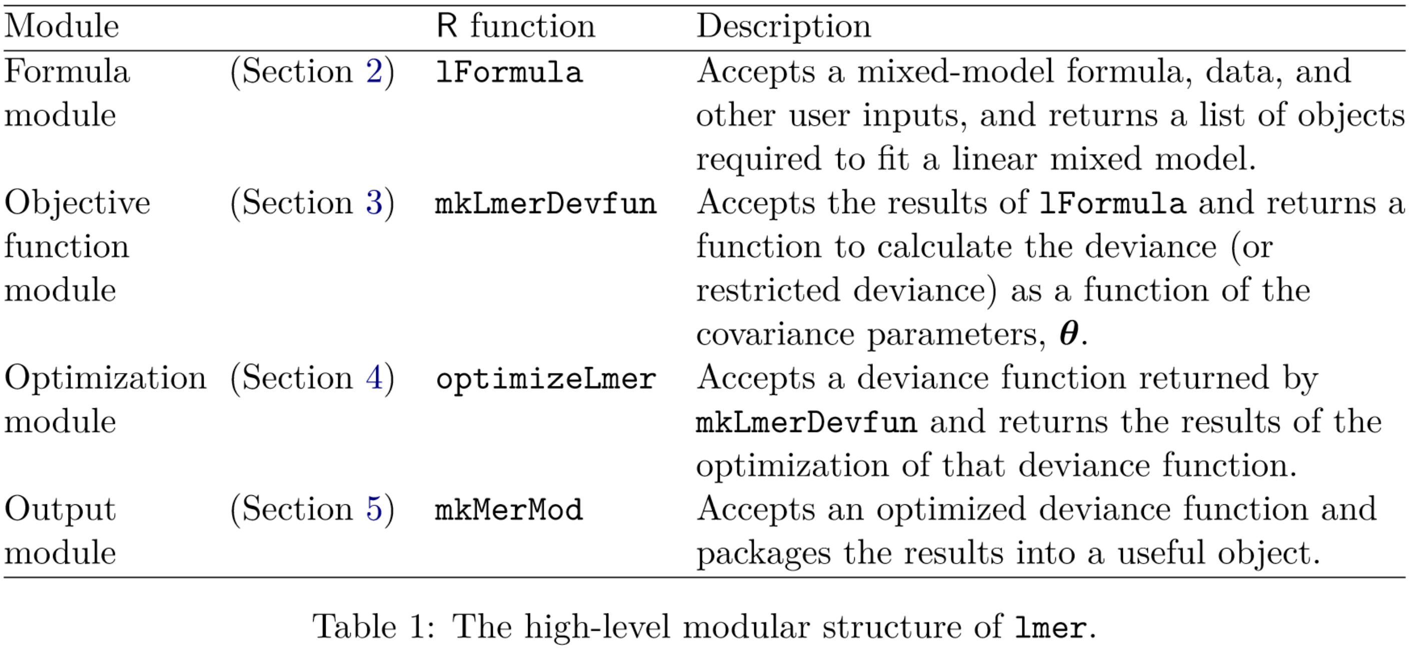
and centre the data using the estimated mean $\hat{\mu}(t)$, i.e. $\tilde{y}_i(t) = y_i(t) - \hat{\mu}(t)$. Then we fit the model on centred data.

Fit Random Effect

The random effect formula used in our genetic model is `( -1 + basis_1 + ... + basis_k | subject id) + ( -1 + basis_1 + ... + basis_k | subject id)` . The first term specifies random slopes for the basis functions grouped by subject id. This means that each individual will have its own random slopes. The second term is identical to the first one, but by including the random-effect term twice, we are essentially fitting two sets of random effects, one for the genetic component and one for the environmental component. This allows us to model the genetic relationships between individuals using the additive genetic relationship matrix $\boldsymbol{A}$.

6. `fit_genetic_fmm()`

Here, we outline how we build the function `fit_genetic_fmm` to fit our genetic mixed-effect model. We use the modularised functions in **lme4** to adjust various steps in fitting genetic data.



Module		R function	Description
Formula module	(Section 2)	`lFormula`	Accepts a mixed-model formula, data, and other user inputs, and returns a list of objects required to fit a linear mixed model.
Objective function module	(Section 3)	`mkLmerDevfun`	Accepts the results of `lFormula` and returns a function to calculate the deviance (or restricted deviance) as a function of the covariance parameters, $\boldsymbol{\theta}$.
Optimization module	(Section 4)	`optimizeLmer`	Accepts a deviance function returned by `mkLmerDevfun` and returns the results of the optimization of that deviance function.
Output module	(Section 5)	`mkMerMod`	Accepts an optimized deviance function and packages the results into a useful object.

Table 1: The high-level modular structure of `lmer`.

1. Formula module: `lFormula` function is used to parse the model formula and to extract the random-effect matrix by `$reTrim$Zt` .
2. Update the upper part of `Z` to incorporate the genetic relation by multiplying matrix $\boldsymbol{M}$. The transpose of the updated `Z` matrix is assigned to `$reTrms$Zt` , effectively replacing the original random effect design matrix.
3. `mkLmerDevfun` function is called with the updated `lFormula` object, and creates an objective function for this genetic model, which is then optimized using the `optimizeLmer` function.
4. `mkMerMod` function is used to create the fitted mixed-effect model object, which is returned by the `fit_genetic_fmm` function.

7. Covariance Functions

Once we have fitted our model, we can extract the estimated genetic covariance matrix $\boldsymbol{C^G}$ using the `VarCorr` method. Then the final step is to convert the covariance matrix to the covariance function by

$$G(s,t) = \boldsymbol{\phi(t)^T} * \boldsymbol{C^G} * \boldsymbol{\phi(t)}$$

Simulation Study

Recall the mixed-effect model:

$$\underbrace{Y_{ij}}_{\text{trait}} = \underbrace{\mu(t_{ij})}_{\text{fixed effect: population mean}} + \underbrace{\sum_{k=1}^{K} \alpha_{ik}\phi(t_{ij})}_{\text{genetic random effect}} + \underbrace{\sum_{k=1}^{K} \gamma_{ik}\phi(t_{ij})}_{\text{environmental random effect}} + \underbrace{\epsilon_{ij}}_{\text{measurement error}}$$

In matrix form:

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Z^G}\alpha + \mathbf{Z^E}\gamma + \epsilon$$

with the following distribution of random vectors:

$$\alpha \sim N(\mathbf{0}, \mathbf{A} \otimes \mathbf{C^G})$$
$$\gamma \sim N(\mathbf{0}, \mathbf{I_N} \otimes \mathbf{C^E})$$
$$\epsilon \sim N(\mathbf{0}, \sigma^{\mathbf{2}}_{\mathbf{res}} * \mathbf{I_n})$$

assuming there are N individuals and total n measurements.

For the simulation study, we will fix a basis $\phi(t)$ for our functional space and true covariance matrices $\mathbf{C^G}$ and $\mathbf{C^E}$. Therefore the true genetic and environmental covariance functions which will be estimated are:

$$G(t,s) = G(s,t) = \boldsymbol{\phi(t)^T} * \boldsymbol{C^G} * \boldsymbol{\phi(t)}$$
$$E(t,s) = G(s,t) = \boldsymbol{\phi(t)^T} * \boldsymbol{C^E} * \boldsymbol{\phi(t)}$$

Then we will generate a set of responses and fit the mixed-effect model to get the estimated covariance functions $\hat{G}(t,s)$ and $\hat{E}(t,s)$, which then will be compared with true ones.

Simulation process

Step 1:set up functional basis and "true" covariances.

We use 5 cubic B-spline as model basis and fix equal covariance for both genetic and environmental covariances. C^G and C^E are 5×5 matrices of correlations between the elements of each α_i and γ_i. Remark: we will use 3 (and 4) principal components as basis to fit the mixed-effect model. It is reasonalbe to assume the number of basis to construct the covariance function is larger than the number of principal components needed to re-construct the covariance.

$$C^G = \begin{pmatrix} 750 & 10 & 130 & 80 & 250 \\ 10 & 800 & 30 & 15 & 40 \\ 130 & 30 & 700 & 50 & 130 \\ 80 & 15 & 50 & 420 & 50 \\ 250 & 40 & 130 & 50 & 330 \end{pmatrix}$$

and

$$C^E = \begin{pmatrix} 750 & 10 & 130 & 80 & 250 \\ 10 & 800 & 30 & 15 & 40 \\ 130 & 30 & 700 & 50 & 130 \\ 80 & 15 & 50 & 420 & 50 \\ 250 & 40 & 130 & 50 & 330 \end{pmatrix}$$

Step 2: Set the distribution of the random effect vectors α and γ.

We assume α and γ follow multivariate normal distribution with covariance $\mathbf{A} \otimes \mathbf{C^G}$ and $\mathbf{I_N} \otimes \mathbf{C^E}$. Remark: we will assume there are 873 individuals so the same genetic relationship matrix $\mathbf{A}$ computed from the actual data can be used in our simulation study. Each individual has 10 regular sampling points from the unit interval. We will later change the number of measurements to see how it will affect the simulation result.

Step 3: Generate curve data.

Each individual data is influenced by a combination of random effects across the basis functions plus the measurement errors. The reason we ignore the fixed effect is that we compute the principal components on centered data when fitting the mixed-effect model.

Step 4: Fit simulated data into our model. We generate 50 groups of responses and measure the errors on genetic and environmental covariance functions for each simulation with L2 and Frobenius norm.