École Polytechnique Fédérale de Lausanne

Averaged Variational Inference for Hierarchical Modelling of Genetic Association

Author:
William van Rooij

Supervisors:
Hélène Ruffieux
Prof. Anthony Davison

A thesis submitted for the degree of

MSc in Applied Mathematics

11th July 2019

Abstract

Expression quantitative trait locus (eQTL) analyses study the effects of genetic variants on the expression of transcripts or genes. The data used generally consist of several hundred thousand genetic variants and thousands of transcript expression outcomes.

In this work, we suppose that the data follow a hierarchical regression model linking the genetic variants and the outcomes. We are then confronted with a *small n, large p, large q* situation, where p is the number of genetic variants, q is the number of expression levels, and n is the number of samples. In this situation, MCMC algorithms are not suitable for Bayesian inference as their computational cost is too large.

Here, we present a fast variational algorithm to estimate the associations between genetic variants and traits based on Ruffieux et al. [2017]. We perform a weighted average of variational estimates obtained from different parameter initialisations and augment our method with simulated annealing.

We evaluate the performance of our proposal by comparing it to existing approaches and assess its accuracy through comparisons with MCMC inference on a small problem.

The code for all our numerical experiments is freely accessible at
https://github.com/WilliamVanRooij/MasterProject.

Contents

Chapter 1

Introduction

1.1 Situation

For the past years, data science has been increasingly present in the world. From financial establishments to road management companies, a lot of industry sectors are integrating data science into the way business is done. With the expansion of computer performance, we are able to implement faster computation and can work with more complex models. The volume of available, hence analysable, data is also growing, which allows more accurate inference.

Often, when trying to find a model for data, we have many more observations than parameters to fit: a *large n, small p* situation. This is the most common type of statistical analysis. Bayesian hierarchical modelling is a valuable tool to identify the dependencies across multiple sources of information, but the number of parameters may be much larger than the number of observations. This is often the case in genomic research, where the situation is called *small n, large p*. Traditional techniques do not apply then, because of both statistical and computational constraints.

In this thesis, we will focus on the *small n, large p* situation in the context of genetic association. We will tackle high-dimensional regression in the Bayesian framework, with its statistical advantages and its computational problem, which often dissuades users from adopting this solution in statistical applications.

1.2 Motivation

Current technology allows us to numerically represent the human genome: a whole new set of data is available to study the influence of the genome on diseases or phenotypes. Some of these newly-available data measure *genetic variants*, changes at specific locations

on the genome (loci), the different versions of which are called *alleles*. We will focus on the most common category of genetic variants, namely, *single nucleotide polymorphisms* (SNPs), i.e., variations in the nucleotides that are present to some appreciable extent in the population. Some combinations of SNPs are inherited together, which yields block-wise dependence structures.

Figure 1.1 shows the correlations between real SNPs, located in region ENm014 on the seventh chromosome , from a Yoruba population [Altshuler et al., 2005]. We clearly see a local block structure; outside the blocks, the correlations are not null but very small. A strong block correlation structure means that two SNPs in the same block may be statistically hard to differentiate. The goal is to represent the probabilities of association between a SNP and a trait of interest, while conveying the uncertainty implied by the block correlation in our results.

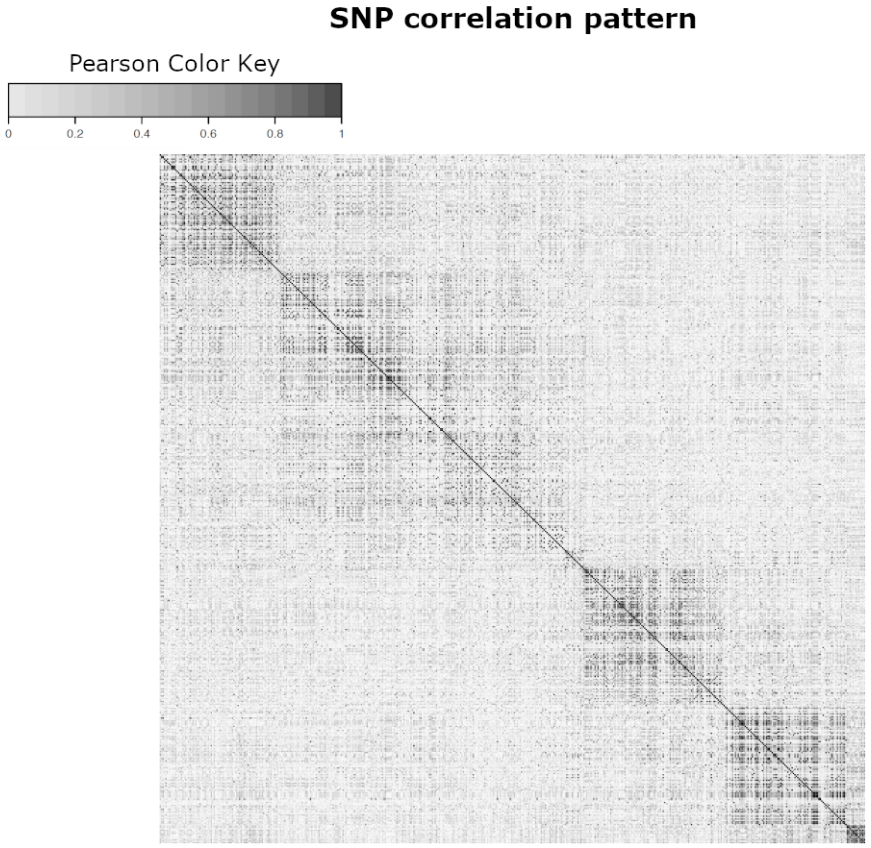


Figure 1.1: Block correlation structure of SNPs taken from a Yoruba population HapMap, ENm014 region, chromosome 7 [Altshuler et al., 2005]. The darker the dot, the stronger the correlation between the two corresponding SNPs.

We focus on *expression quantitative trait locus* (eQTL) analyses, which study the effects of genetic variants, in our case SNPs, on the expression of transcripts or genes. The data used for eQTL studies generally consist of several hundred thousand SNPs and thousands of expression outcomes. It is, in fact, a *small n, large p, large q* situation, where p is the number of SNPs, q is the number of expression outcomes, and n is the number of samples.

Bayesian inference involves many integrals, which usually need to be approximated. Markov Chain Monte Carlo (MCMC) algorithms are a standard technique for the approximation of integrals and can be fast and accurate when working on reasonably small datasets. When the dataset dimensions grow, however, MCMC algorithms tend to become very time-consuming. Indeed, when performing MCMC inference, likelihoods and sometimes gradients typically need to be calculated at each iteration, and the cost of these calculations increases with the number of parameters. Moreover, the higher the dimension, the less accurate the approximations, and more iterations are needed to reach a given precision. For the algorithm to end, all the parameters need to have converged, meaning that they all need to be checked and stored, which is often impossible when their number is very high.

In our situation, *small n, large p, large q*, the computational cost of using an MCMC algorithm is huge. The time and memory needed to run the algorithm are not acceptable. We have to use an alternative solution, which we choose to be variational inference Blei et al. [2017].

Chapter 2

Hierarchical sparse regression for multiple responses

2.1 Model

Let $\boldsymbol{X} = (X_1, \ldots, X_p)$ be a centered design matrix, representing the candidate predictor SNPs, and $\boldsymbol{y} = (y_1, \ldots, y_q)$ be a centered response matrix, representing the traits. We consider a hierarchical model, where each response y_t is linearly related with the predictors $\boldsymbol{X}$ and has a residual precision τ_t, i.e.,

$$\boldsymbol{y}_{n\times q} = \boldsymbol{X}_{n\times p}\, \boldsymbol{\beta}_{p\times q} + \boldsymbol{\epsilon}_{n\times q}, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \tau_t^{-1} I_n),$$

where $\boldsymbol{\beta}$ is the matrix of regression coefficients. The parameters τ_t and σ^{-2} are assigned Gamma priors.

We introduce $\boldsymbol{\gamma}_{p\times q}$, a binary matrix to indicate which pairs of SNPs and traits are associated. The SNP s and trait t are associated if and only if $\gamma_{st} = 1$. To enforce sparsity on $\boldsymbol{\beta}$, we set a "spike-and-slab" prior distribution on β_{st}, i.e.,

$$\beta_{st} \mid \gamma_{st}, \sigma^2, \tau_t \sim \gamma_{st}\, \mathcal{N}(0, \sigma^2 \tau_t^{-1}) + (1 - \gamma_{st})\, \delta_0,$$

where δ_0 is the Dirac distribution.

The prior distribution of γ_{st} is

$$\gamma_{st} \mid \omega_s \sim \text{Bernoulli}(\omega_s),$$

where the parameter ω_s controls to the proportion of responses associated with the predictor $\boldsymbol{X}_s$, and follows a Beta distribution,

$$\omega_s \sim \text{Beta}(a_s, b_s),$$

with parameters a_s and b_s chosen to enforce sparsity. If we assume $p^* \ll p$, an expected number of predictors involved in the model, we set a_s and b_s such that the prior probability that $\boldsymbol{X}_s$ is associated with at least one response is equal to p^*/p. As we fix the mean of the distribution but let the variance be free, there is still one degree of freedom, so multiple choices are possible, e.g.,

$$a_s = 1,\ b_s = q(p - p^*)/p^*,$$

as in Castillo et al. [2015].

Parameters σ and ω_s are shared across all q traits, which enables the borrowing of strength across all traits having predictors in common.

2.2 Parameters of interest for variable selection

We are interested in estimating the associations between the SNPs and the traits by obtaining summaries of the posterior distribution of $\boldsymbol{\gamma}$ or $\boldsymbol{\beta}$, e.g., for the latter,

$$\begin{aligned} p(\boldsymbol{\beta} \mid \boldsymbol{y}) &= \int \cdots \int p(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\omega}, \boldsymbol{\tau}, \sigma^{-2} \mid \boldsymbol{y})\, \mathrm{d}\boldsymbol{\gamma}\, \mathrm{d}\boldsymbol{\omega}\, \mathrm{d}\boldsymbol{\tau}\, \mathrm{d}\sigma^{-2} \\ &= \frac{1}{p(\boldsymbol{y})} \int \cdots \int p(\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\omega}, \boldsymbol{\tau}, \sigma^{-2})\, \mathrm{d}\boldsymbol{\gamma}\, \mathrm{d}\boldsymbol{\omega}\, \mathrm{d}\boldsymbol{\tau}\, \mathrm{d}\sigma^{-2}, \end{aligned}$$

with

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\omega}, \boldsymbol{\tau}, \sigma^{-2}) = &\left\{ \prod_{t=1}^{q} p(\boldsymbol{y}_t \mid \boldsymbol{\beta}_t, \tau_t) \right\} \left\{ \prod_{t=1}^{q} \prod_{s=1}^{p} p(\beta_{st} \mid \gamma_{st}, \tau_t, \sigma^{-2}) \right\} \\ &\times \left\{ \prod_{t=1}^{q} \prod_{s=1}^{p} p(\gamma_{st} \mid \omega_s) \right\} \left\{ \prod_{s=1}^{p} p(\omega_s) \right\} \left\{ \prod_{t=1}^{q} p(\tau_t) \right\} p(\sigma^{-2}), \end{aligned}$$

where, as mentioned earlier,

$$\begin{aligned} \boldsymbol{y}_t \mid \boldsymbol{\beta}_t, \tau_t &\sim \mathcal{N}_n \left(\boldsymbol{X}\boldsymbol{\beta}_t, \tau_t^{-1} \boldsymbol{I}_n \right), \\ \beta_{st} \mid \gamma_{st}, \tau_t, \sigma^{-2} &\sim \gamma_{st} \mathcal{N} \left(0, \sigma^2 \tau_t^{-1} \right) + (1 - \gamma_{st}) \delta_0, \\ \gamma_{st} \mid \omega_s &\sim \text{Bernoulli}(\omega_s), \\ \omega_s &\sim \text{Beta}(a_s, b_s), \\ \tau_t &\sim \text{Gamma}(\eta_t, \kappa_t), \\ \sigma^{-2} &\sim \text{Gamma}(\lambda, \nu), \end{aligned}$$

for $t = 1, \ldots, q$, $s = 1, \ldots p$, and δ_0 is the Dirac distribution.

Chapter 3

Variational Inference

3.1 General principles

Variational inference simplifies the estimation of the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ by approximating it with a simpler density $q(\boldsymbol{\theta})$ in an optimisation problem that minimizes a measure of "closeness". More precisely, given a family of densities $\mathcal{D}$ over the parameters, we want to find the distribution $q \in \mathcal{D}$ that is the closest to $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ in terms of the Kullback–Leibler divergence

$$\mathrm{KL}(q \parallel p) := \int q(\boldsymbol{\theta}) \log\left(\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \mid \boldsymbol{y})}\right) \mathrm{d}\boldsymbol{\theta}.$$

This divergence was introduced in 1951 by Kullback and Leibler [1951] and is the most common divergence measure used in statistics and machine learning. It is described as a "directed divergence" as it is asymmetric, i.e.,

$$\mathrm{KL}(p \parallel q) \neq \mathrm{KL}(q \parallel p).$$

Choosing the family $\mathcal{D}$ can be difficult, as we need it to be simple enough to enable tractable inference, but flexible enough for q to accurately represent $p(\boldsymbol{\theta} \mid \boldsymbol{y})$. The approximation will then be

$$q^*(\boldsymbol{\theta}) = \underset{q(\boldsymbol{\theta}) \in \mathcal{D}}{\arg\min}\, \mathrm{KL}\left[q(\boldsymbol{\theta}) \parallel p(\boldsymbol{\theta} \mid \boldsymbol{y})\right].$$

As its expression involves the marginal likelihood, directly minimizing the Kullback–Leibler divergence can be complicated, depending on the density p that we want to approximate and the density family $\mathcal{D}$ that we want q to be part of. For this reason, we decompose the Kullback–Leibler divergence as

$$\begin{aligned}\mathrm{KL}\left[q(\boldsymbol{\theta}) || p(\boldsymbol{\theta} \mid \boldsymbol{y})\right] &= \mathbb{E}\left[\log q(\boldsymbol{\theta})\right] - \mathbb{E}\left[\log p(\boldsymbol{\theta} \mid \boldsymbol{y})\right] \\ &= \mathbb{E}\left[\log q(\boldsymbol{\theta})\right] - \mathbb{E}\left[\log p(\boldsymbol{y}, \boldsymbol{\theta})\right] + \log p(\boldsymbol{y}),\end{aligned}$$

and introduce the "evidence lower bound" on the marginal log-likelihood:

$$\mathcal{L}(q) = \mathbb{E}\left[\log p(\boldsymbol{\theta}, \boldsymbol{y})\right] - \mathbb{E}\left[\log q(\boldsymbol{\theta})\right] = \int q(\boldsymbol{\theta}) \log \frac{p(\boldsymbol{y}, \boldsymbol{\theta})}{q(\boldsymbol{\theta})} \mathrm{d}\boldsymbol{\theta},$$

i.e., we obtain

$$\mathrm{KL}(q \parallel p) = \log(p) - \mathcal{L}(q).$$

This means that the Kullback–Leibler divergence is the difference between the marginal log-likelihood, with no effect on the optimisation, and a function $\mathcal{L}(q)$. Hence, minimizing the Kullback–Leibler divergence is the same as maximizing $\mathcal{L}(q)$. The difference lies in the complexity of the problems: minimizing the Kullback–Leibler divergence is typically not tractable, but maximizing $\mathcal{L}(q)$ admits a closed form when the family of densities $\mathcal{D}$ is well chosen. For this reason, variational inference uses $\mathcal{L}(q)$ as its objective function.

Jensen's inequality provides another way to see that $\mathcal{L}(q)$ is a lower bound for the marginal log-likelihood,

$$\begin{aligned}
\log p(\boldsymbol{y}) &= \log \int p(\boldsymbol{y}, \boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \\
&= \log \int \frac{p(\boldsymbol{y}, \boldsymbol{\theta})}{q(\boldsymbol{\theta})} q(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} \\
&\geq \int q(\boldsymbol{\theta}) \log \left\{ \frac{p(\boldsymbol{y}, \boldsymbol{\theta})}{q(\boldsymbol{\theta})} \right\} \mathrm{d}\boldsymbol{\theta} \\
&= \mathcal{L}(q).
\end{aligned}$$

Hence, $\log p(\boldsymbol{y}) \geq \mathcal{L}(q)$.

3.2 Mean-field approximation

The complexity of the optimisation problem is directly bound to the complexity of the family of densities $\mathcal{D}$ to which $q(\boldsymbol{\theta})$ belongs. We introduce the mean-field variational family, where the parameters are mutually independent a posteriori, i.e., let $\{\theta_j\}_{j=1}^{J}$ be a partition of $\boldsymbol{\theta}$. Then,

$$q(\boldsymbol{\theta}) = \prod_{j=1}^{J} q_j(\theta_j).$$

We determine the variational factors $q_j(\theta_j)$ by maximizing $\mathcal{L}(q)$. Hence, the variational family does not directly represent the observed data, they are linked through the optimisation of the evidence lower bound.

In our case, we assume the posterior independence of most of the parameters,

$$q(\boldsymbol{\theta}) = \left\{ \prod_{s=1}^{p} \prod_{t=1}^{q} q(\beta_{st}, \gamma_{st}) \right\} \left\{ \prod_{s=1}^{p} q(\omega_s) \right\} \left\{ \prod_{t=1}^{q} q(\tau_t) \right\} q(\sigma^{-2});$$

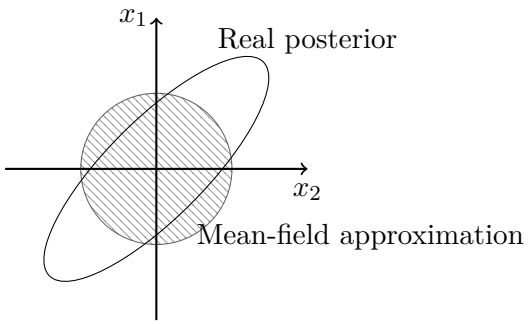


Figure 3.1: Example of the mean-field approximation, for a two-dimensional Gaussian distribution (in clear). The mean-field approximation of the posterior distribution is represented by the barred circle. The mean of the approximation agrees with the real mean, but the covariance does not match the covariance of the real posterior.

we keep β_{st} and γ_{st} grouped in order to obtain a "spike-and-slab" form a posteriori for each of the factors, rather than unimodal distributions, which would ignore the multimodal behaviour induced by the spike-and-slab prior.

We have transformed, using the evidence lower bound and the mean-field approximation, our problem into a optimisation problem. We now need a way to solve this problem. In the following section, we describe the coordinate ascent algorithm.

3.3 Coordinate ascent

The coordinate ascent algorithm is typically used to solve the optimisation problem arising in mean-field variational inference. It iterates on the variational parameters of the mean-field approximation, optimising them one at the time and yields a local optimum for the evidence lower bound. The algorithm is based on the following result:

Lemma 3.1 *If we fix $q_l(\theta_l)$, $l \neq j$, then the optimal $q_j^*(\theta_j)$ satisfies*

$$q_j^*(\theta_j) \propto \exp\left\{\mathbb{E}_{-j}\left[\log p(\theta_j \mid \boldsymbol{\theta}_{-j}, \boldsymbol{y})\right]\right\},$$

where $\mathbb{E}_{-j}$ denotes the expectation with respect to all θ_l, $l \neq j$.

Based on this result, the algorithm updates one parameter θ_j at a time while the others stay fixed. The algorithm stops when $\mathcal{L}(q)$ increases by less than a pre-determined tolerance ε.

At every iteration, $\mathcal{L}(q)$ is guaranteed to increase. The local optimum thus obtained may depend on the initialization of the $q_j(\theta_j)$, $j = 1, \ldots, J$; different initializations could yield different optima that correspond to different models.

Algorithm 1: Coordinate ascent variational inference

input : $p(\boldsymbol{y}, \boldsymbol{\theta})$, dataset y, tolerance ε

output : $q(\boldsymbol{\theta}) = \prod_{j=1}^{J} q_j(\theta_j)$

initialize: the parameters of each $q(\theta_j)$

repeat

  for $j \in \{1, \ldots, J\}$ **do**

    set $q_j(\theta_j) \propto \exp\left\{\mathbb{E}_{-j}\left[\log p(\theta_j \mid \boldsymbol{\theta}_{-j}, \boldsymbol{y})\right]\right\}$

  $\mathcal{L}^{\mathsf{old}}(q) \leftarrow \mathcal{L}(q)$

  $\mathcal{L}(q) \leftarrow \mathbb{E}\left[\log p(\boldsymbol{\theta}, \boldsymbol{y})\right] - \mathbb{E}\left[\log q(\boldsymbol{\theta})\right]$

until $|\mathcal{L}^{\mathsf{old}}(q) - \mathcal{L}(q)| < \varepsilon$

return $q(\boldsymbol{\theta})$

For our model, the posterior distributions of our model parameters are:

$$
\begin{aligned}
\beta_{st} \mid \gamma_{st} = 1, \boldsymbol{y} &\sim \mathcal{N}\left(\mu_{\beta,st}, \sigma^2_{\beta,st}\right), \\
\beta_{st} \mid \gamma_{st} = 0, \boldsymbol{y} &\sim \delta_0, \\
\gamma_{st} \mid \boldsymbol{y} &\sim \text{Bernoulli}(\gamma^{(1)}_{st}), \\
\omega_s \mid \boldsymbol{y} &\sim \text{Beta}(a^*_s, b^*_s), \\
\tau_t \mid \boldsymbol{y} &\sim \text{Gamma}(\eta^*_t, \kappa^*_t), \\
\sigma^{-2} \mid \boldsymbol{y} &\sim \text{Gamma}(\lambda^*, \nu^*),
\end{aligned}
$$

for $s = 1, \ldots, p$, $t = 1, \ldots, q$, where $\mu_{\beta,st}$, $\sigma^2_{\beta,st}$, $\gamma^{(1)}_{st}$, a^*_s, b^*_s, η^*_t, κ^*_t, λ^*, and ν^* are the "variational" parameters obtained after convergence of Algorithm 1. Their complete expression is given in Appendix B of Ruffieux [2018b].

Chapter 4

Multimodality

4.1 Problem statement

When applied to highly correlated data, variational inference underestimates posterior variances, as explained in Blei et al. [2017]. Suppose that $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ is the posterior distribution of $\boldsymbol{\theta} = (\theta_1, \theta_2)$ and that we use the mean-field approximation

$$q(\boldsymbol{\theta}) = q(\theta_1)q(\theta_2).$$

As we can see in Figure 3.1, the covariance structure is altered (θ_1 and θ_2 are independent a posteriori) and the marginal variances are smaller than those of $p(\boldsymbol{\theta} \mid \boldsymbol{y})$. This also results from the optimisation of the reverse Kullback–Leibler divergence

$$\mathrm{KL}(q \parallel p) = -\int q(\boldsymbol{\theta}) \log \frac{p((\theta \mid \boldsymbol{y})}{q(\boldsymbol{\theta})} \mathrm{d}\boldsymbol{\theta},$$

which penalizes putting mass in $q(\cdot)$ where $p(\cdot)$ has little mass.

The lower bound $\mathcal{L}(q)$ tends to be highly multimodal, so the ascent algorithm (Algorithm 1) risks to get stuck in local modes. The posterior variance underestimation reinforces this risk, putting a lot of mass on one single hypothesis.

To handle this multimodality better, we will explore two routes to enhance variational inference, without changing the model. The first is to introduce a simulated annealing procedure to explore more modes; this was proposed by Ruffieux et al. [2018]. The second is to average over multiple parameter initialisations with weights equal to the posterior model probability corresponding to the obtained mode. We describe these two options in Sections 4.2 and 4.3.

4.2 Annealed variational inference

Simulated annealing aims at improving the exploration of multimodal parameter spaces, using heated distributions to sweep the local modes away and ease the progression to the global mode. We next describe how it can be coupled with variational inference.

We start with the same strategy as in Section 3.1, i.e., minimising the reverse Kullback–Leibler divergence,

$$\mathrm{KL}(q \parallel q) = -\int q(\boldsymbol{\theta}) \log \left\{ \frac{p(\boldsymbol{\theta} \mid \boldsymbol{y})}{q(\boldsymbol{\theta})} \right\} \mathrm{d}\boldsymbol{\theta},$$

and use the lower bound evidence as objective function,

$$\mathcal{L}(q) = \mathbb{E}_q \left[\log p(\boldsymbol{y}, \boldsymbol{\theta}) \right] - \mathbb{E}_q \left[\log q(\boldsymbol{\theta}) \right].$$

The objective function is composed of the expected log joint distribution, which implies that the approximation will put more mass where the variables best explain the data, and the entropy, which encourages the "dispersion" of the approximation.

The idea of simulated annealing is to introduce a temperature T to obtain a series of heated distributions,

$$p_T(\boldsymbol{y}, \boldsymbol{\theta}) \propto p(\boldsymbol{y}, \boldsymbol{\theta})^{1/T},$$

and control the "frequency" of the modes. The temperature starts high, smoothing the density of interest, and gets lower along the process until the original density is reached. The high temperatures facilitate the search for the global optimum. The temperature multiplies the entropy term, allowing for more disperse approximations

$$\mathcal{L}_T(q_T) = \int q_T(\boldsymbol{\theta}) \log p(\boldsymbol{y}, \boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta} - T \int q_T(\boldsymbol{\theta}) \log q_T(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}, \quad T \geq 1, \tag{4.1}$$

where q_T is the heated variational distribution. Hence, annealed variational inference applies a penalty on the log joint distribution when the temperature $T > 1$, and relaxes the penalty as T goes down until $T = 1$, where the penalty becomes null.

To obtain the annealed variational factors, $q_T(\theta_j)$, we write (4.1) with respect to θ_j as

$$\begin{aligned} \mathcal{L}_T(q) &= \mathbb{E}_j \left[\mathbb{E}_{-j} \left\{ \log p(\boldsymbol{y}, \boldsymbol{\theta}) \right\} - T \log q_T(\theta_j) \right] + \text{const} \\ &= T \mathbb{E}_j \left[\log \left\{ \frac{p_{T,-j}(\boldsymbol{y}, \theta_j)}{q_T(\theta_j)} \right\} \right] + \text{const}, \end{aligned}$$

where $p_{T,-j}(\boldsymbol{y}, \theta_j) \propto \exp \left\{ T^{-1} \mathbb{E}_{-j} \left[\log p(\boldsymbol{y}, \boldsymbol{\theta}) \right] \right\}$, $\mathbb{E}_j$ is the expected value with respect to $q_T(\theta_j)$, $\mathbb{E}_{-j}$ is the expected value with respect to every $q_T(\theta_k)$ where $k \neq j$, and const is independent of θ_j. The objective for $\mathcal{L}_T(q)$ is maximal when $q_T(\theta_j) = p_{T,-j}(\boldsymbol{y}, \theta_j)$, i.e., when

$$\log q_T(\theta_j) = T^{-1} \mathbb{E}_{-j} \left[\log p(\boldsymbol{y}, \boldsymbol{\theta}) \right] + \text{const}, \quad j = 1, \ldots, J.$$

Different choices are possible for the temperature schedule, including geometric spacing,

$$T_l = (1 + \Delta)^{l-1}, \quad \Delta = T_L^{1/(L-1)} - 1,$$

harmonic spacing,

$$T_l = 1 + \Delta(l-1), \quad \Delta = \frac{T_L - 1}{L-1},$$

and linear spacing,

$$T_l^{-1} = T_L^{-1} + \Delta(L-l), \quad \Delta = \frac{1 - T_L^{-1}}{L-1},$$

where $l = 1, \ldots, L$ and T_L is the hottest temperature. T_l is the temperature used at step l and L is the number of steps used to lower the temperature to the initial temperature $T = 1$. The original variational algorithm is then run until convergence.

4.3 Averaged variational inference

Bayesian model averaging is a strategy to account for multiple competing models in an inference problem. It consists of weighting the different models in a weighted average, accounting for the likelihood that the data corresponds to each model. The more the model corresponds to the observed data, the more it will stand out in the result.

Assume that the data $\boldsymbol{y}$ may have been obtained from one of multiple models M_k, $k = 1, \ldots, K$, and Δ is the quantity of interest. The posterior distribution

$$p(\Delta \mid \boldsymbol{y}) = \sum_{k=1}^{K} p(\Delta \mid M_k, \boldsymbol{y})\, p(M_k \mid \boldsymbol{y}) \tag{4.2}$$

corresponds to a weighted average of the posterior distribution under each of the considered models with weights corresponding to the posterior model probabilities. Instead of $p(\Delta \mid \boldsymbol{y})$ in (4.2), we might be interested in summaries like the posterior mean

$$\mathbb{E}\left[\Delta \mid \boldsymbol{y}\right] = \sum_{k=1}^{K} \mathbb{E}\left[\Delta \mid M_k, \boldsymbol{y}\right]\, p(M_k \mid \boldsymbol{y}).$$

The posterior probability for model M_k is given by

$$p(M_k \mid \boldsymbol{y}) = \frac{p(\boldsymbol{y} \mid M_k)\, p(M_k)}{\sum_{j=1}^{K} p(\boldsymbol{y} \mid M_j)\, p(M_j)}, \tag{4.3}$$

where $p(\boldsymbol{y} \mid M_k)$ is the likelihood under model M_k, and $p(M_k)$ is the prior probability of model M_k. This may, for example, be chosen based on the model complexity, to favour simpler models, or, if we consider the models to be a priori equiprobable, it is set to $p(M_k) = 1/K$, $k = 1, \ldots, K$.

In Section 3.1, we saw that the evidence lower bound and the Kullback–Leibler divergence are related,

$$\mathrm{KL}(q \parallel p) = \log p(\boldsymbol{y}) - \mathcal{L}(q),$$

and that minimizing the Kullback–Leibler divergence is equivalent to maximizing the evidence lower bound. Hence, by assuming that $\mathcal{L}(q)$ is a tight lower bound on the marginal log likelihood, we can use it as an approximation for $\log p(\boldsymbol{y} \mid M_k)$ in (4.3).

We propose to address the concerns described in Section 3.1 by performing a form of averaging of variational inference summaries. Namely, say that our quantity of interest is γ_{st}, to assess the association between SNP s and trait t. Using Algorithm 1, we initialise the distributions $q_j(\theta_j)$ with different starting points, and consider the optimums yielded by the algorithm. If we consider that each optimum yields a model representing the data, we can apply an averaging procedure to combine them all using the method described above. We approximate $\log p(\boldsymbol{y})$ by $\mathcal{L}(q)$ in (4.3), and obtain an approximation for $\mathbb{E}[\gamma_{st} \mid \boldsymbol{y}]$ considering all the models obtained through the algorithm.

To cope with the high multimodality induced by strongly correlated structures and represent the uncertainty of the modes, we use simulated annealing combined with our weighted averaging procedure and retrieve a combination of different models yielded from different initialisations. We hope that the uncertainty in the selected variables will be conveyed in the resulting approximations for $\mathbb{E}[\gamma_{st} \mid \boldsymbol{y}]$.

Chapter 5

Simulations

5.1 Preliminary illustration

In this chapter, we assess the performance of our averaged variational method on simulations. We use the `locus` `R`-package [Ruffieux, 2019] and call the variational algorithm multiple times before combining all the results in a weighted average. As explained in Section 4.3, we initialise the parameters differently for each call, in order to possibly obtain different optima. Then we use the evidence lower bound of the different calls as weights to combine the posterior summaries of each initialisation.

For all simulations presented in this chapter, we simulate data with very strong correlation patterns to evaluate the benefit of our method in the extreme multimodality scenarios it is designed for.

We use the `echoseq` `R`-package [Ruffieux, 2018a] to generate blocks of strongly autocorrelated SNPs and traits, as well as associations between them. The SNPs are coded as discrete variables describing their state and we create dependence between them using realisations of multivariate normal variables followed by a quantile thresholding rule.

For our first illustration, we generate 300 observations of 500 SNPs, by blocks of 10 SNPs, with latent variable block autocorrelations between 0.95 and 0.99. For simplicity, we simulate just one trait; the extension to multiple traits should produce similar conclusions. We select five SNPs to be associated with the trait and, for better visualisation, all five SNPs are among the 50 first SNPs.

Figure 5.1 shows the probabilities of association of the 50 first SNPs, out of 500 used: the LOCUS method is equivalent to choosing a single model M and calculating

$$\mathbb{E}\left[\gamma_{st} \mid \boldsymbol{y}\right] = \mathbb{E}\left[\gamma_{st} \mid M, \boldsymbol{y}\right] \; p\left(M \mid \boldsymbol{y}\right).$$

Our "averaged LOCUS" method uses a weighted average over 100 different initialisations

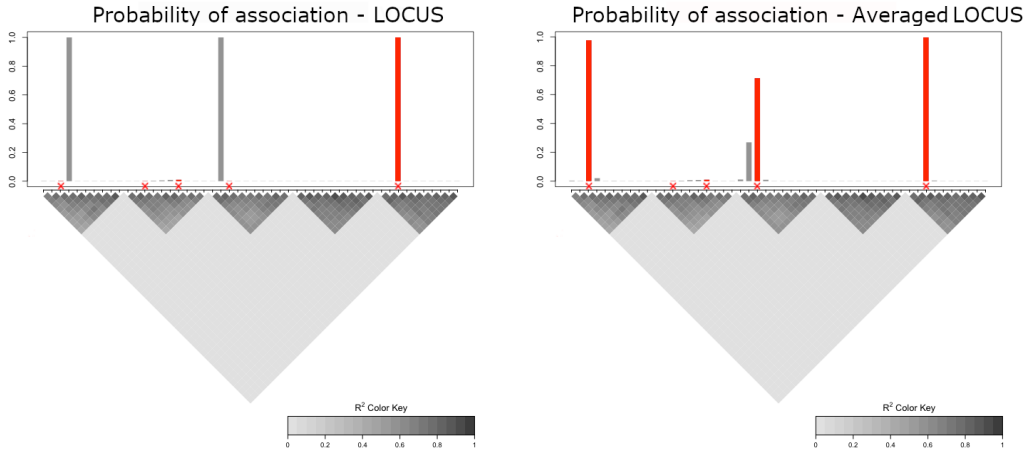


Figure 5.1: Probabilities of association of the 50 first SNPs with a single trait estimated using the original LOCUS method (left) and using our "averaged LOCUS" proposal (right), which implements the weighted averaged described in Section 4.3. In red are the five SNPs simulated as associated with the response, they are also marked with a red cross. Underneath are the extreme correlation patterns of the SNPs; they are the same for the two sides as the SNPs used are the same.

yielding 100 models M_k, $k = 1, \ldots, 100$:

$$\mathbb{E}\left[\gamma_{st} \mid \boldsymbol{y}\right] = \sum_{k=1}^{100} \mathbb{E}\left[\gamma_{st} \mid M_k\right] \; p\left(M_k \mid \boldsymbol{y}\right).$$

With the original LOCUS method, the algorithm wrongly selects two SNPs and misses four SNPs simulated as associated with the response. This can be explained by the strong correlations in the block structure creating a highly multimodal posterior and misleading the algorithm: it selected wrong SNPs among strongly correlated SNPs.

Our averaged variational inference algorithm does better; it identifies three of the five relevant SNPs. It also better conveys the block correlation structure in the probabilities of association as four SNPs of the middle block all have non null probabilities of association with the trait.

5.2 Variable selection performance

In this section, we compare four methods: classical variational inference (LOCUS), averaged variational inference (averaged LOCUS) and their simulated annealing augmented counterparts (annealed LOCUS and averaged annealed LOCUS). We choose four different situations: two of the settings involve 15 associated SNPs (settings A, B), whereas the

remaining two have 50 associated SNPs (settings C, D). For simplicity, we consider only one trait. For a pair of settings, the proportion of the response variance explained by the SNPs is below 50% (settings A, C) and, for another pair, below 80% (settings B, D). The simulated annealing augmented methods have an initial temperature fixed at $T_L = 2$, and a geometric spacing with ten steps. The sensitivity to these choices could be assessed in dedicated experiments. The remaining settings are the same as for Figure 5.1.

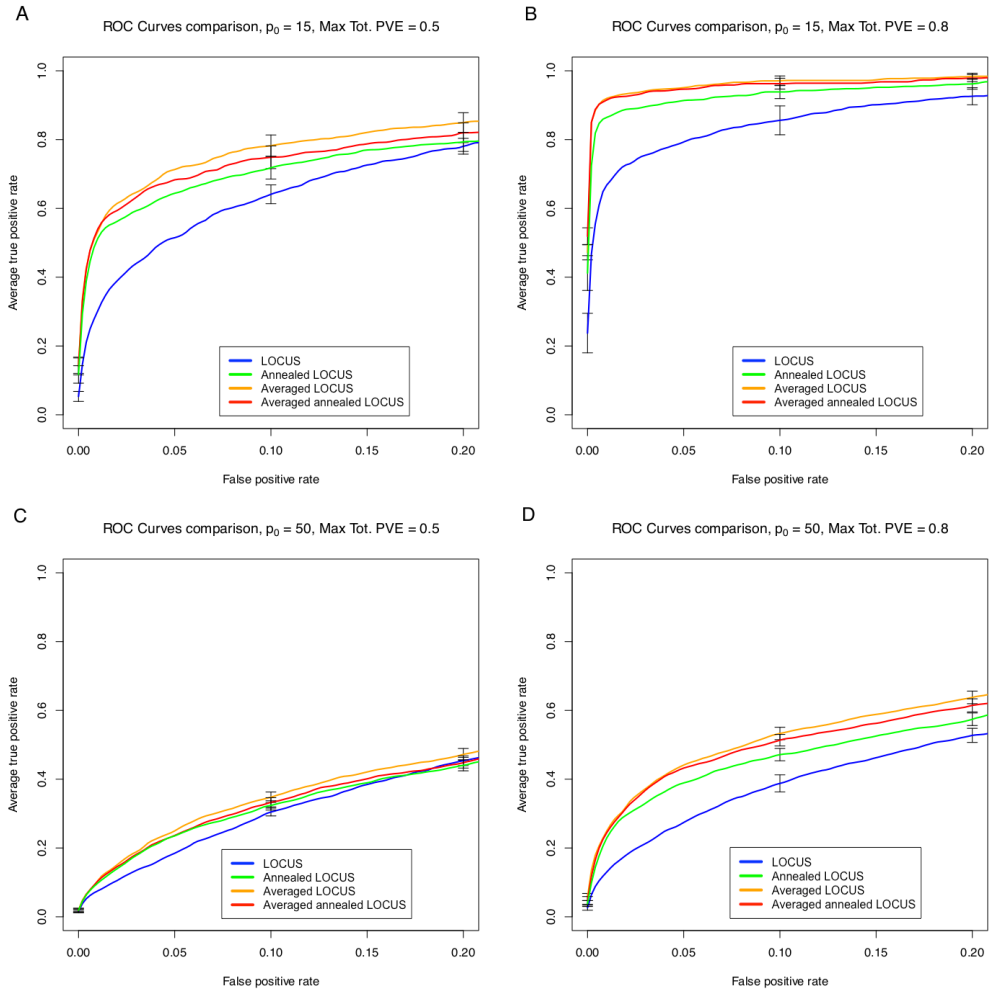
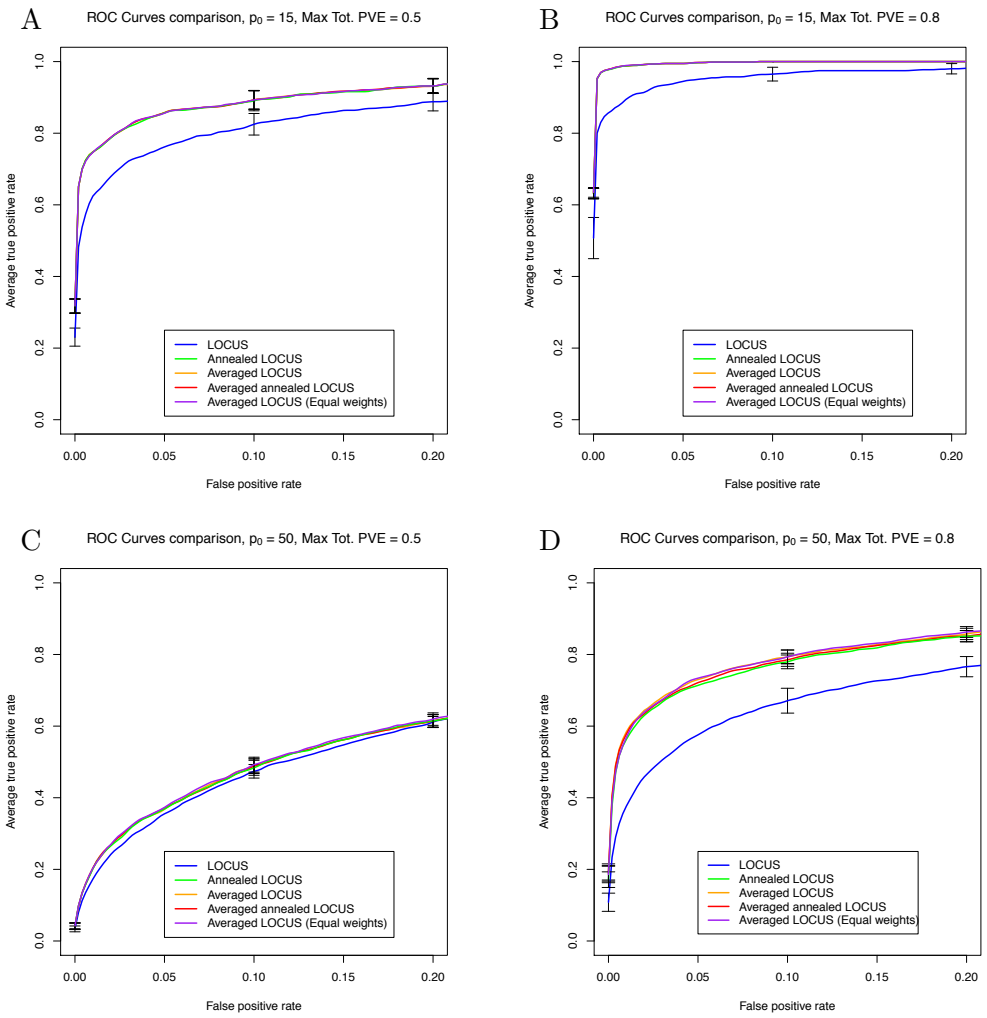


Figure 5.2: Comparison of ROC curves between LOCUS, averaged LOCUS, and the same two methods augmented with a simulated annealing step, colored orange, blue, red, and green respectively. Top row: $p_0 = 15$, Left column: Max tot. PVE= 0.5, Bottom row: $p_0 = 50$, Right column: Max tot. PVE= 0.8

Figure 5.2 shows the variable selection performance in terms of ROC curves for the four methods, for each of the four settings. We truncate the ROC curves, as we are interested only in the performance of the methods for small false positive rate.

First, the averaged LOCUS method clearly outperforms the LOCUS method in all four



scenarios: it seems that the weighted averaging procedure effectively alleviates the risk of selecting wrong predictors in groups of highly correlated SNPs.

Second, when starting both LOCUS and averaged LOCUS with a simulated annealing step, averaged LOCUS continues to be more powerful than LOCUS, although the improvement is smaller than without simulated annealing. This suggests that the annealing step does not prevent the averaged LOCUS algorithm from selecting multiple different models, in this strongly correlated data scenario. This could be because the chosen initial temperature is not sufficiently high to smooth the densities enough to access the right modes.

Third, annealed LOCUS outperforms LOCUS. The simulated annealing step allows the method to reach modes that cannot be reached by the LOCUS method with certain starting parameters.

Fourth, in the less sparse setting with 50% of variance explained by the predictors (setting C), the simulated effect sizes are weaker and all methods show similar, lower, performances: the averaging or annealing procedures do not lead to much improvement.

Finally, averaged annealed LOCUS performs similarly to averaged LOCUS: their confidence intervals overlap. In setting A, averaged annealed LOCUS might even be less powerful: the simulated annealing step might diminish the number of modes considered for the average, putting more weight on wrong models.

5.3 Comparison with MCMC inference

Section 5.2 evaluated variable selection performance of the different methods, we now compare the accuracy of our proposal by confronting it with MCMC inference. To do so, we generate data with the `echoseq` R-package, and save the simulated matrix $\boldsymbol{\beta}$. We simulate 300 observations for equicorrelated SNPs with extremely high correlation coefficient of 0.955.

We compare the posterior distributions of the regression coefficient obtained by our methods with the posterior distributions obtained by MCMC inference. The two inference methods have a different convergence and stopping criteria, so the comparison should be studied prudently. Our method is based on variational inference, which has a convergence criterion defined as a tolerance to be given. The MCMC inference does not necessarily visit the whole model space, so to alleviate this problem, we run it for a large number of iterations, namely 10^5 iterations and discard the first half as burn-in, and we consider a very small problem, i.e., $p = 4, q = 1$. We are interested in evaluating the posterior distributions of $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3, \beta_4)$. In the construction of our data, we have chosen $\beta_2, \beta_3 = 0$ and $\beta_1, \beta_4 \neq 0$.

Figure 5.3 shows LOCUS and averaged LOCUS estimated posteriors of $\boldsymbol{\beta}$, as well as the histogram of the MCMC posteriors and the simulated values of $\boldsymbol{\beta}$.

First, the problem appears to be very difficult as all methods disagree to some extent and fail to accurately capture the simulated values; even the MCMC algorithm yields inferences far from the truth, particularly for β_1 and β_4.

Second, averaged LOCUS probably best reflects the true posterior; it puts mass near the simulated values of β_s for every β_s but for β_4, where it finds the same estimation as the MCMC inference and the LOCUS methods. This is in line with the ROC curves of Figure 5.2, where we saw that for variable selection, averaged LOCUS outperforms LOCUS.

Third, when LOCUS and averaged LOCUS disagree, the result of LOCUS is "visible" in the distribution of averaged LOCUS. Averaged LOCUS considers the mode obtained from LOCUS in its averaging.

Finally, β_4 is supposed to be non-null, but the MCMC approximations and those given

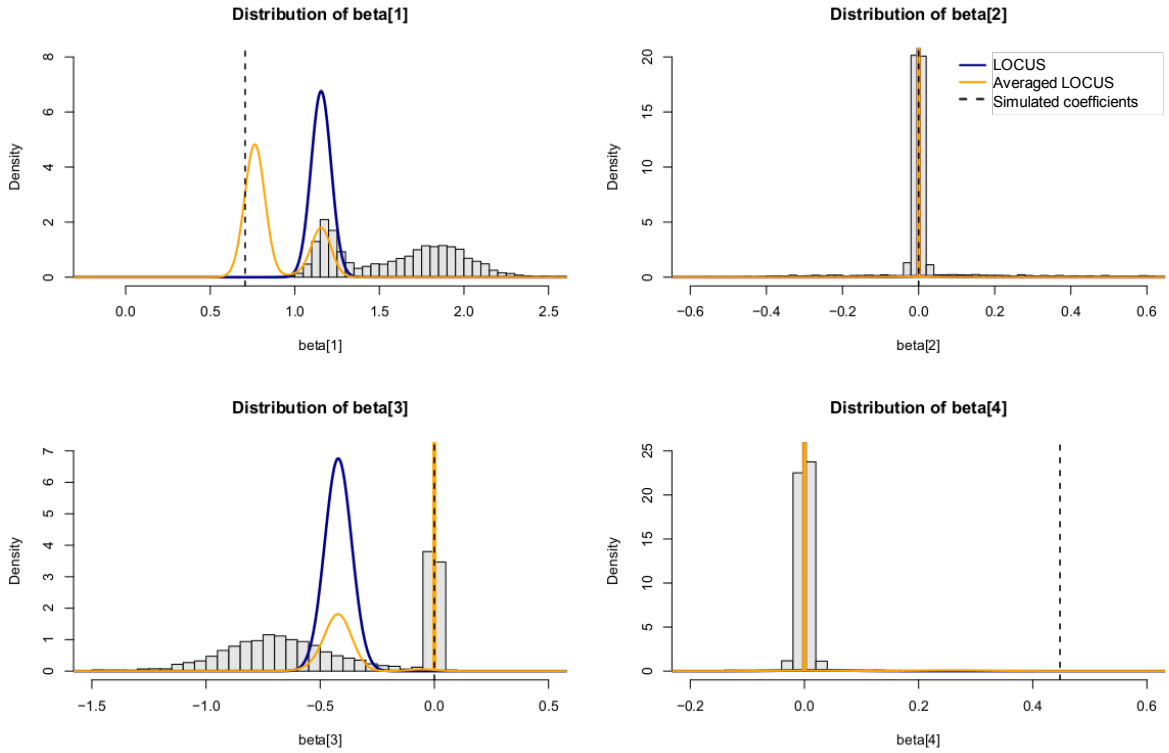


Figure 5.3: Comparison of LOCUS (blue) and averaged LOCUS (orange) estimated posterior distributions for $\boldsymbol{\beta}$, MCMC distributions (histograms) as well as the simulated $\boldsymbol{\beta}$ values (dashed black line). The orange and blue lines of β_2 and β_4 are superimposed.

by LOCUS and averaged LOCUS are all concentrated around zero. The strong correlation gave the wrong mode too much weight, giving the illusion that it was the global mode. This can be an effect of the spike-and-slab prior which enforces too much shrinkage.

Figure 5.4 shows the same posteriors as Figure 5.3, but with a simulated annealing step added to the LOCUS and averaged LOCUS methods. We have used the same settings than for Figure 5.3, so the histograms and the simulated $\boldsymbol{\beta}$ values are the same for the two situations. We chose an initial temperature $T_L = 5$, and used ten geometric steps.

For all four β_s, annealed LOCUS yields a posterior density that is more aligned with averaged annealed LOCUS. The posterior given by annealed LOCUS tends to put mass at the same place than the averaged annealed LOCUS posterior.

As for the standard methods, the simulated annealing augmented methods overlap the simulated values for all β_s except for β_4 where, the MCMC simulation as well as the augmented methods yield a posterior with values concentrated around zero.

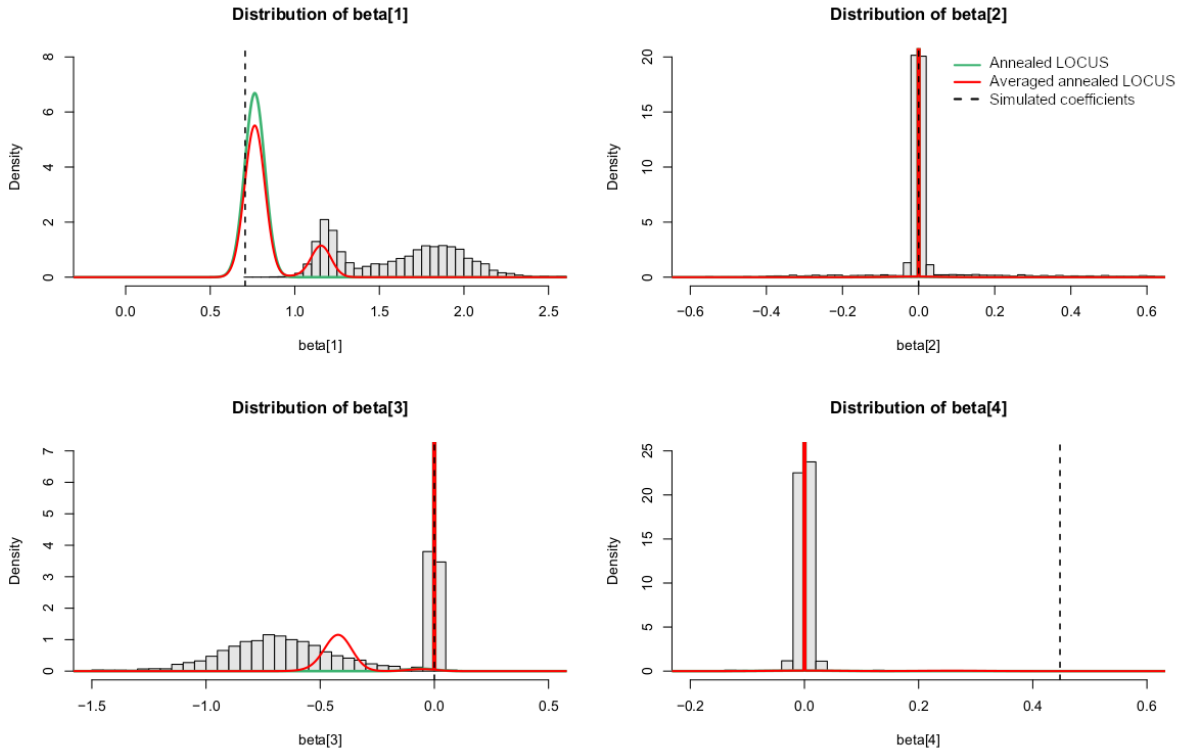


Figure 5.4: Comparison of annealed LOCUS (green) and averaged annealed LOCUS (red) estimated posterior distribution for $\boldsymbol{\beta}$, MCMC distributions (histograms) $\boldsymbol{\beta}$ posteriors as well as the simulated $\boldsymbol{\beta}$ values (dashed black line).

When comparing the plots of Figures 5.3 and 5.4, one sees that the annealing changed the posterior densities. In Figure 5.3, the posterior density of β_1 and β_3 were on a wrong mode, but in Figure 5.4 they overlap the simulated $\boldsymbol{\beta}$.

5.4 Running times

Our method, whether with simulated annealing or not, can be implemented in parallel, which tends to drastically diminish the runtime. Even if the method has to wait until the last run to converge, we would still be quicker than calculating the runs one after the other.

Figure 5.5 shows the running times of the methods, computed on 500 SNPs, 15 SNPs associated with a single trait, and for the averaged versions, 100 different initialisations, measured for 50 replications. For comparison, the runtime of calculating 100 times the

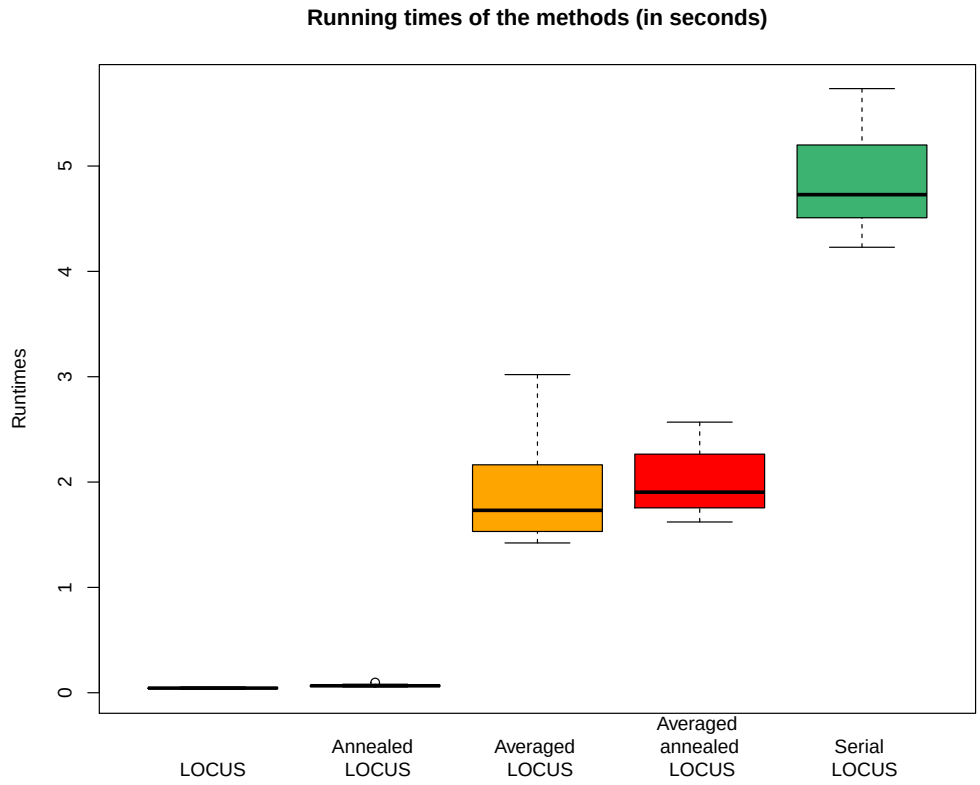


Figure 5.5: Running times, in seconds, of the methods: LOCUS (first), annealed LOCUS (second), averaged LOCUS (third), averaged annealed LOCUS (fourth), and 100 iterations of LOCUS (fifth), computed on 300 observations of 500 SNPs. 15 SNPs are associated with a single trait, and for the averaged versions, 100 different initialisations, averaged over 50 replications. The SNPs are correlated in blocks of ten with an autocorrelation between 0.98 and 0.99. Up to 50% of the response variance is explained by the SNPs.

LOCUS method is shown on the same figure. The advantage of paralleled implementation is highlighted as it takes approximately twice as much time to compute 100 iterations of the LOCUS method, one after another, than to compute the averaged LOCUS method.

The averaged annealed LOCUS takes the longest to compute out of the first four methods, as on addition to averaging, the method starts every occurrence with an annealing step. The annealed LOCUS also takes more time to compute than the standard LOCUS method, which confirms the additional time needed for the averaged annealing LOCUS to complete, compared to the averaged LOCUS.

Chapter 6

Conclusion

In this work, we proposed a new variational approach based on Bayesian averaging to efficiently deal with strong data correlation in genetic association problems.

We compared the variable selection performance, posterior distributions, and runtime of four methods: the original variational implementation from the package `locus` (LOCUS), our weighted average augmented method (averaged LOCUS), and their simulated annealing augmented counterparts (annealed LOCUS and averaged annealed LOCUS).

Our proposal, averaged LOCUS, helps better convey the uncertainty implied by strong block correlation structures in genetic data. It also outperforms the original LOCUS method is terms of variable selection. Moreover, the annealed LOCUS method performs better than the LOCUS method, but the averaged annealed LOCUS performs similarly to the averaged LOCUS method. Finally, LOCUS is faster than averaged LOCUS but parallel computation is possible, so the runtimes can be greatly reduced.

Several improvements may be considered. First, we also need to evaluate the method performance when considering more that one trait, i.e., $q > 1$. Real eQTL data are made of more than one trait so it would better assess the performance of the methods on relevant data.

Second, we assumed that every model is a priori equiprobable in the averaged LOCUS procedure; other choices could be considered. For example, we could relate the model probability with the expected number of associated SNPs.

Third, for the annealing procedure, we have chosen a geometric schedule, a number of steps L, and an initial temperature T_L. It would be good to compare the performance with different choices of initial temperatures, steps, and schedule.

Fourth, we would like to apply this method on real eQTL data.

Finally, provided that all our experiences show good performance, we will optimise our

code and integrate it in the `locus` package (http://github.com/hruffieux/locus).

Bibliography

Altshuler, D., Donnelly, P., and International HapMap Consortium (2005). A haplotype map of the human genome. *Nature*, 437:1299.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112:859–877.

Castillo, I., Schmidt-Hieber, J., and van der Vaart, A. (2015). Bayesian linear regression with sparse priors. *Annals of Statistics*, 43.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22:79–86.

Ruffieux, H. (2018a). *echoseq: Replication and simulation of genetic variants, molecular expression levels and other phenotypic data*. R package version 0.2.3.

Ruffieux, H. (2018b). *Large-scale variational inference for Bayesian joint regression modelling of high-dimensional genetic data*. PhD thesis, EPFL.

Ruffieux, H. (2019). *locus: Large-scale variational inference for combined selection of covariate and response variables in regression models*. R package version 0.9.0.

Ruffieux, H., Davison, A., Hager, J., Inshaw, J., Fairfax, B., Richardson, S., and Bottolo, L. (2018). A global-local approach for detecting hotspots in multiple-response regression.

Ruffieux, H., Davison, A. C., Hager, J., and Irincheeva, I. (2017). Efficient inference for genetic association studies with multiple outcomes. *Biostatistics*, 18:618–636.