Averaged Variational Inference for Hierarchical Modelling of Genetic Association

William van Rooij
École Polytechnique Fédérale de Lausanne
July 2019

Introduction

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. However, in genomic research, the number of parameters is often much larger than the number of observations, the situation is called *small n, large p*. Traditional techniques do not apply then, because of both statistical and computational constraints. We will focus on this 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.
Current technology allows us to 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.

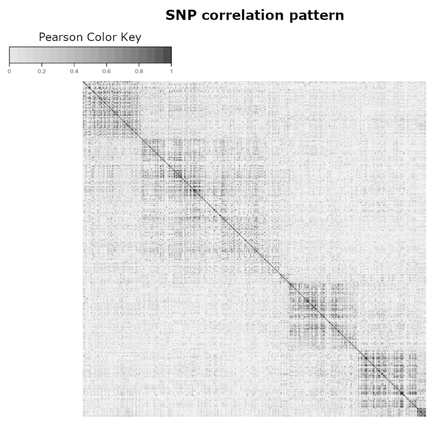

Figure: Block correlation structure of SNPs taken from a Yoruba population HapMap, ENm014 region, chromosome 7 [Altshuler et al., 2005]. The darker the dot, the stronger the correlation between the two corresponding SNPs.

Figure 1 shows the correlations between real SNPs, located in region ENm014 on the seventh chromosome , from a Yoruba population . 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.
We focus on *expression quantitative trait locus* (eQTL) analyses, that 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. 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].

Hierarchical sparse regression for multiple responses

Let $\boldsymbol{X} = (X_1, \dots, X_p)$ be a centered design matrix, representing the candidate predictor SNPs, and $\boldsymbol{y} = (y_1, \dots, y_q)$ 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.,

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

where β is the matrix of regression coefficients. The parameters τ_t and σ^{-2} are assigned Gamma priors.
We introduce $\gamma_{p\times q}$, 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 β, 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.
We are interested in estimating the associations between the SNPs and the traits by obtaining summaries of the posterior distribution of γ or β.

Variational Inference

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

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

This divergence was introduced in 1951 by Kullback and Leibler [1951] and is the most common divergence measure used in statistics and machine learning.
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 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

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

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.

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 $\mathcal{L}(q)$. In our case, we assume the posterior independence of most of the parameters,

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

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.

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
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.

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 $j \in \{1, \dots, J\}$ **do**
    set $q_j(\theta_j) \propto \exp\left\{\mathbb{E}_{-j}\left[\log p(\theta_j \mid \boldsymbol{\theta}_{-j}, \boldsymbol{y})\right]\right\}$
  $\mathcal{L}^{\text{old}}(q) \leftarrow \mathcal{L}(q)$
  $\mathcal{L}(q) \leftarrow \mathbb{E}\left[\log p(\boldsymbol{\theta}, \boldsymbol{y})\right] - \mathbb{E}\left[\log q(\boldsymbol{\theta})\right]$
until $|\mathcal{L}^{old}(q) - \mathcal{L}(q)| < \varepsilon$;
return $q(\boldsymbol{\theta})$

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

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

for $s = 1, \dots, p$, $t = 1, \dots, q$, where $\mu_{\beta,st}$, $\sigma^2_{\beta,st}$, $\gamma_{st}^{(1)}$, a_s^*, b_s^*, η_t^*, κ_t^*, λ^*, and ν^* are the "variational" parameters obtained after convergence of Algorithm 1.

Problem statement

When applied to highly correlated data, variational inference underestimates posterior variances, as explained in Blei et al. [2017].
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.

Annealed variational inference

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

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

where q_T is the heated variational distribution.
The objective for $\mathcal{L}_T(q)$ is maximal when $q_T(\theta_j) = p_{T,-j}(\boldsymbol{y}, \theta_j)$, i.e., when

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

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

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

where $l = 1, \dots, L$ 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.

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, $k = 1, \dots, K$, and Δ is the quantity of interest. The posterior distribution

$$\mathbb{E}\left[\Delta \mid \boldsymbol{y}\right] = \sum_{k=1}^{K} \mathbb{E}\left[\Delta \mid M_k, \boldsymbol{y}\right]\; p(M_k \mid \boldsymbol{y}). \tag{2}$$

corresponds to a weighted average of the posterior distribution under each of the considered models with weights corresponding to the posterior model probabilities.
The posterior probability for model M_k is given by

$$p(M_k \mid \boldsymbol{y}) = \frac{p(\boldsymbol{y} \mid M_k)\; p(M_k)}{\sum_{j=1}^{K} p(\boldsymbol{y} \mid M_j)\; p(M_j)}, \tag{3}$$

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.
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 (3).
We perform 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 (3), and obtain an approximation for $\mathbb{E}\left[\gamma_{st} \mid \boldsymbol{y}\right]$ considering all the models obtained through the algorithm.

Preliminary illustration

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 **??**, 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.
We simulate data with very strong correlation patterns to evaluate the benefit of our method in the extreme multimodality scenarios it is designed for.
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.

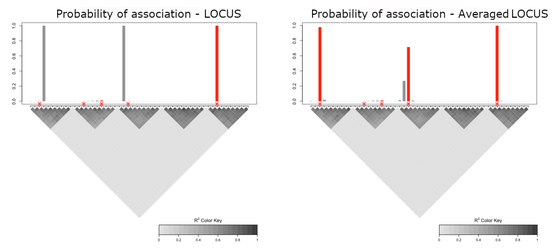

Figure: Probabilities of association of the 50 first SNPs with a single trait estimated using the original LOCUS method (left) and using our "averaged LOCUS" proposal (right), which implements the weighted averaged described in Section **??**. In red are the five SNPs simulated as associated with the 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.

Figure 2 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, where our "averaged LOCUS" method uses a weighted average over 100 different initialisations yielding 100 models.
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.

Variable selection performance

We now compare four methods: classical variational inference (LOCUS), averaged variational inference (averaged LOCUS) and their simulated annealing augmented counterparts (annealed LOCUS and averaged annealed LOCUS). The simulated annealing augmented methods have an initial temperature fixed at $T_L = 2$, and a geometric spacing with ten steps.

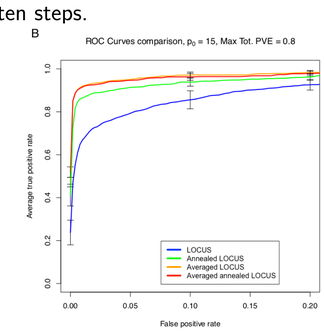

Figure: Comparison of ROC curves between LOCUS, averaged LOCUS, and the same two methods augmented with a simulated annealing step, colored orange, blue, red, and green respectively. The data involve 15 SNPs associated with one trait, for simplicity, and the response variance is explained by the SNPs is below 80%.

Figure 3 shows the variable selection performance in terms of ROC curves for the four methods. 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: 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.
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.
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.

Comparison with MCMC inference

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 β. 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 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 $\beta_1, \beta_4 \neq 0$.

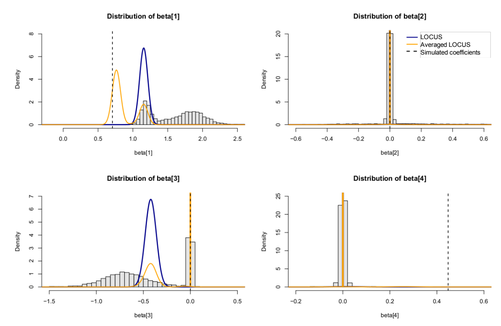

Figure: Comparison of LOCUS (blue) and averaged LOCUS (orange) estimated posterior distributions for β, MCMC distributions (histograms) as well as the simulated β values (dashed black line). The orange and blue lines of β_2 and β_4 are superimposed.

Figure 4 shows LOCUS and averaged LOCUS estimated posteriors of β, as well as the histogram of the MCMC posteriors and the simulated values of β.
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 3, 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 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.

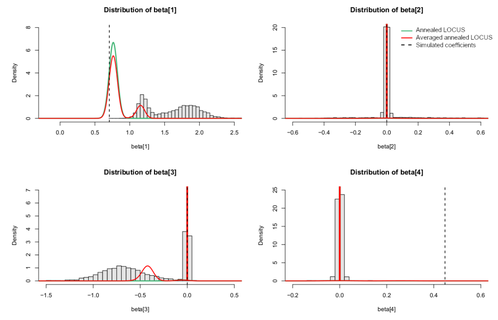

Figure: Comparison of annealed LOCUS (green) and averaged annealed LOCUS (red) estimated posterior distribution for β, MCMC distributions (histograms) β posteriors as well as the simulated β values (dashed black line).

Figure 5 shows the same posteriors as Figure 4, but with a simulated annealing step added to the LOCUS and averaged LOCUS methods. We have used the same settings than for Figure 4, so the histograms and the simulated β 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.
When comparing the plots of Figures 4 and 5, one sees that the annealing changed the posterior densities. In Figure 4, the posterior density of β_1 and β_3 were on a wrong mode, but in Figure 5 they overlap the simulated β.

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.

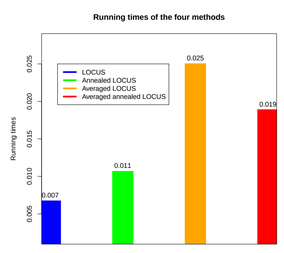

Figure: Running times, in seconds, of the four methods: LOCUS (blue), annealed LOCUS (green), averaged LOCUS (orange), and averaged annealed LOCUS (red), computed on 500 SNPs, a single trait, and for the averaged versions, 100 different initialisations, averaged over 20 replications.

Figure 6 shows the running times of the four methods, computed on 500 SNPs, a single trait, and for the averaged versions, 100 different initialisations, averaged over 20 replications. The two averaged methods take more time than the two others, which is expected, but knowing they each are made of 100 initialisations highlights the efficiency of the parallel implementation.
The averaged LOCUS method has a longer runtime than the averaged annealed LOCUS method, the convergence of the averaged annealed LOCUS is probably reached earlier thanks to the annealing procedure.

References

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.

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

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.