Genotype to phenotype analysis on the scale of millions of human genomes

Authors

Hamed Heydari[1,2,*], Changjiang Xu[1], Charles Boone[1,2,3*], Gary D. Bader[1,2,4,5,6,7*]

Affiliations

1. The Donnelly Centre, University of Toronto, Ontario, Canada
2. Department of Molecular Genetics, University of Toronto, Ontario, Canada
3. RIKEN Center for Sustainable Resource Science, Wako, Saitama, Japan
4. Department of Computer Science, University of Toronto, Toronto, Ontario, Canada
5. The Lunenfeld-Tanenbaum Research Institute, Sinai Health System, Toronto, Ontario, Canada
6. Princess Margaret Cancer Centre, University Health Network, Toronto, Ontario, Canada
7. CIFAR Multiscale Human Program, CIFAR, Toronto, Ontario, Canada

* Correspondence: h.heydari@mail.utoronto.ca, gary.bader@utoronto.ca, charlie.boone@utoronto.ca

Abstract

Linear mixed models (LMMs) are the most-established model of the genotype to phenotype relationship in populations of individuals, but they do not scale to the level of hundreds of thousands of genotyped individuals in large human biobanks. We developed a scalable framework for fitting likelihood-based multi-component LMMs with full-rank covariance that scales to several million samples with high statistical accuracy, rivaling exact computation, without the high computational and memory demands of previous state-of-the-art methods. Our scalable LMM (SLMM) implementation can be distributed from one to many computers aiding fast and accurate large-scale analysis. We applied SLMM to examine ~300,000 individuals and estimate heritability, infer selection-related parameters, and incorporate prior knowledge of specific genome components, such as codons, gene promoters and terminators, as well as biological pathway gene sets to perform functional enrichment on several phenotypes from individual data in the UK Biobank.

Main Text

Human genetics seeks to answer three important questions based on information about how genotypes and phenotypes vary across populations of individuals: “How heritable is a trait?”, “What is the biological mechanism of heritability?” and “How does an individual’s genotype predict their phenotype?”. Genotype and phenotype information for over 20 million human individuals has been, or will be, collected by biobanks such as UK Biobank[1], Our Future Health[2], the Million Veteran Program[3], All of Us[4], CanPath[5], FinnGen[6], Biobank Japan[7], Taiwan Biobank[8] and Shanghai Zhangjiang Biobank[9], as well as by companies like 23andMe[10]. Analysis of human biobank information provides immense opportunities to explore human genetic variation along with its complex interaction structure, both internally and with the environment, and their impact on human health[11]. It also opens a new frontier for genetics, as analogous data have not been established for any other organism.

Accurate and thorough human genetic analysis is challenging because the genotype to phenotype mapping is confounded by complex human population structure and other factors. Further, state-of-the-art statistical genetics methods to handle these challenges do not scale to the large sizes of human biobanks. The linear mixed model (LMM) is a flexible statistical approach that combines fixed and random effects to analyze hierarchical or clustered data[12] and is a prominent solution to this challenge. It can model the relationship between predictor variables and an outcome variable while accounting for both population-level effects (fixed effects) and group-specific or subject-specific variations (random effects) with diverse application across wide variety of scientific fields[13]. In genetics, LMMs have been used extensively for estimating heritability of phenotypic variation[14–17], inferring the role of selection[18,19], variance partitioning[20], rare and common variant association analysis[21–26]. Restricted maximum likelihood (REML) is the unbiased and statistically efficient method for fitting LMMs[12,13]. The major disadvantage of current REML implementations[24,26,27] is that they do not scale well to large sample sizes (e.g. hundreds of thousands of individuals), preventing their use on large human biobanks, as they suffer from cubic complexity in sample size. Approximate LMM solutions have been developed to increase speed but do so by compromising on one or more of the following important statistical properties: statistical efficiency[28], full-rank estimation[22,23], likelihood evaluation[21,28,29], or model design versatility.

We developed scalable LMM (SLMM), an efficient and accurate approach for performing likelihood-based estimation of variance components that scales to several million samples. Inspired by recent advances in machine learning[30–32], SLMM contributes key methodological improvements that enable it to quickly and accurately solve LMMs on extremely large datasets. We also developed a versatile software implementation of SLMM that can run on any computer from a laptop to a large distributed system, such as a cluster or cloud, and use small or large memory resources. SLMM enables accessible analysis of the largest human biobanks and will lead to more accurate and comprehensive mapping of the genotype to phenotype relationship.

SLMM methodological novelties

Solving LMMs is traditionally slow as it depends on intensive algebraic operations, including matrix inversion, trace, and log-determinant (see Methods). To perform these computations in a scalable manner, we combined several approaches and techniques from machine learning, statistics, numerical methods and software engineering:

1. Complete avoidance of matrix inversion by using highly efficient state-of-the-art Krylov-based approaches to solve linear systems (Block Lanczos[35] and Deflated Preconditioned Block Conjugate Gradient[36]). This transforms our problem into a series of matrix vector products, allowing matrix-free operation, see Methods, Supplementary Figure 1.

2. Application of Hutchinson's randomized trace estimation[34] to convert trace calculation involving matrix inverse (needed for the gradient) into solving a set of linear systems, employing the aforementioned solvers. Variance reduction[33] is then used to minimize the variance of this unbiased estimation procedure, approaching exact calculation. See Methods and supplementary Figures 2 & 3.

3. Reuse of information from the solvers to enable accurate and unbiased estimates of the log determinant and, consequently, the likelihood, through the use of stochastic Lanczos quadrature[35,37,38].

4. Acceleration of the Hessian/information matrix calculation through information reuse from the gradient evaluation step (i.e., block Lanczos tridiagonal matrices) as a warmup strategy. Using

second-order derivatives speeds fitting. We use trust region optimization[39] to aid global convergence.

5. To further expedite the fitting process, we implemented pretraining on a subset of data to derive a better starting point, thereby reducing computation on the full dataset.

6. Given that the model fitting is transformed into a series of linear systems requiring only matrix vector products, we focused on optimizing multiplication with genotype data by implementing an efficient version of the Mailman method[40,41] (see Methods). This enables matrix multiplication in sub-linear time complexity with genotype data consisting of limited possible values. We demonstrate that this multiplication routine is faster than the best general-purpose multiplication methods[42] (e.g. Intel(R) oneAPI Math Kernel Library version 2022.1) (Supplementary Figure 4).

7. Development of a versatile software implementation of SLMM capable of running on diverse computer types, from laptops to large distributed systems such as clusters or clouds, and using small or large memory resources (native support for in-core, out-of-core and distributed operation). See Figure 1 and Supplementary Figure 5).

Together, these advances in mathematical formulation of the REML problem and algorithmic implementations combine to shift the computational complexity of accurate REML solving from cubic to close to linear for SLMM (Figure 1). Since statistical power of genetic analysis increases as the number of individuals analyzed grows, we anticipate that our advance applied to the genetic analysis of millions of individuals will lead to increased discovery of genotype to phenotype relationships and a reduction in false positive results.

SLMM accuracy and runtime performance

Genetics research relies on LMMs perhaps more than any other field[43]. A major reason for this is that LMMs can effectively control for population structure and other confounding factors, which is critical to genetic data interpretation. Statistical geneticists have developed some of the leading LMM solvers and applications, such as the GCTA[27] software for exact REML fitting. To evaluate the accuracy of SLMM compared to the current state-of-the-art exact models in this context, we used GCTA software as a reference ‘exact solution’ approach and performed extensive simulations

on genetic data. GCTA works efficiently with relatively small sample sizes and so we initially focussed analysis on 10,000 individuals derived from the UK Biobank (see Methods).

SLMM provides highly accurate and unbiased estimates, comparable to the exact LMM model (Supplementary Figures 6 and 7) with little loss of statistical efficiency (Supplementary Figure 8). To evaluate the accuracy and convergence properties of SLMM on large data, we performed another round of simulation on a larger subset of 300,000 UK Biobank data (see Methods). Our findings show that SLMM can accurately compute the correct variance parameters and converge successfully across different scenarios (Supplementary Figure 9).

To illustrate the scalability and speed of SLMM, we performed four complementary analyses. First, we compared the runtime of SLMM to that of GCTA as a state-of-the-art exact LMM solver and show that SLMM is an order of magnitude faster than GCTA (Figure 1a). Second, to investigate the speedup of running SLMM across more than one computer, we ran a representative large-scale analysis across an increasing number of computers and measured the relative speedup (Figure 1b) which shows that SLMM has near linear scalability with respect to an increase in the number of machines used. We also show that our approach is faster than BOLT-LMM, a representative scalable approach of the type that approximates the solution by trading off accuracy for increased speed (Supplementary Figure 10). BOLT-LMM does not compute likelihood, only supports genetic random effects with fixed variant-level weights, exhibits bias (Supplementary Figure 6) and is not parallelizable across computers. Third, we show that SLMM can operate out-of-core with little memory requirement, while still maintaining scalability (Supplementary Figure 5). Finally, as we did not have access to data larger than the UK Biobank, to illustrate the scalability of SLMM, we simulated genotypes of 5 million individuals (see Methods), each with 400 thousand variants and were able to fit a multi-component REML (two components, even and odd chromosomes) on such data in 6 hours using 5 compute nodes. To the best of our knowledge, this is the largest reported analysis using a REML-based LMM.

Large-scale heritability analysis

To test SLMM on a real analysis problem, we used it to estimate heritability of quantitative traits in the UK Biobank. We focused on implementing state-of-the-art heritability models (see Methods), such as GCTA's GREML (single component[14], LDMS-I[15] and LDMS-R[44]) and LDAK

(LDAK[17,18] and LDAK-thin[45]) that provide the most accurate trait heritability estimates, some of which support a variety of genetic architectures (e.g. minor allele frequency (MAF) and linkage disequilibrium (LD) dependent architectures)[15,46]. It has not previously been possible to evaluate how well these models fit biobank scale data to estimate trait heritability from individual-level data[15,18,45] and how well their heritability estimates agree, because no LMM solver has existed to fit them to biobank data while providing accurate estimates along with likelihood for comparison.

We analyzed several traits, including benchmark phenotypes such as height and BMI along with a number of blood biomarkers (Table 1, Supplementary Figures 11 to 14), on biobank-scale data using an average of about 330 thousand individuals and 5 million high-quality imputed SNPs (see Methods). We also fitted a more complex model based on the best overall model (GCTA LDMS-I), but including 27 additional components representing various functional categories of genomic regions (e.g. promoters, coding regions). Our findings revealed a considerable difference in the heritability estimate of various models (Figure 2a and Supplementary Figures 11 and 12) along with their likelihood (Supplementary Figures 13 and 14) showing that even at such large sample sizes the estimates don't converge to similar values and therefore the choice of heritability model is important. Our analysis confirms the finding of previous studies that the genetic architecture of traits is highly dependent on the MAF and LD of variants (Supplementary Figure 15), meaning that the distribution of variant effect sizes varies across the MAF and LD spectrum. Moreover, we find that incorporating functional annotation significantly improves the fit of the model according to all reported criteria which supports the use of more complex heritability models using diverse genomic annotation, as observed in small scale studies[37]. Finally, we find GREML-LDMS-I to be a superior choice compared to both LDAK and LDAK-thin according to all reported measures across all phenotypes, which is in contrast to that of recently reported analysis that performed a similar comparison using an approximate likelihood analysis based on summary statistics[45].

As summary statistic-based approaches have recently become popular for estimating heritability[47], we estimated heritability using both summary statistics with LD score regression (LDSC)[48] and individual level data using REML computed using SLMM, as a gold standard[15,49–51]. Our findings demonstrate that REML-based approaches are more accurate and confident than LDSC with large data sets (Supplementary Figure 16), which agrees with previous reports testing this with smaller data sets[15,28,46,47].

Negative selection of heritable human traits

As a second application (Figure 2b), we used SLMM to find signatures of negative selection in 23 quantitative human traits in the UK Biobank by inferring the relationship between SNP MAF and heritability by estimating the selection-related parameter, alpha, of the LDAK-thin model (see Methods, Supplementary Figure 17)[18,19,45]. Negative alpha estimates indicate less common trait-related SNPs tend to have larger effect sizes and vice versa. We find that alpha is generally negative and shows a large degree of variation across all phenotypes, with blood LDL and cholesterol levels being under strongest negative selection.

Genome annotation and heritability analysis

We next used SLMM to perform functional heritability enrichment analysis, where the contribution of functional subsets of SNPs in explaining heritability variance is compared to variance explained by all the SNPs (see Methods). We included 27 functional SNP annotation groups (e.g. non-synonymous SNPs, SNPs in conserved regions) as additional components in the best heritability model (GREML-LDMS-I). The model includes 47 components (including 20 standard LD and MAF related components) and 20 million SNP-to-component mappings (many SNPs map to multiple components) representing approximately 300 thousand rows (individuals) and 20 million columns (SNPs) (Figure 2c and Supplementary Figure 18). Overall, functional enrichments exhibit considerable differences across phenotypes. For example, while non-synonymous SNPs are the most enriched category for height, it is not the top enrichment for BMI (a related phenotype) where variants that are conserved among primates show the highest enrichment. Cholesterol and LDL exhibit higher levels of enrichment for non-synonymous variants, which agrees with our negative selection analysis above, as non-synonymous variants are expected to be under strong negative selection. We also find that variants within core essential genes show statistically significant enrichment for a wide range of phenotypes with higher enrichments for blood glucose and triglycerides levels. Finally, the noticeable difference in enrichment patterns for two functional categories representing enhancers (Enhancer (Fantom5) vs. Enhancer (Hoffman)) shows how important the definition of functional modules and assignment of SNPs to each category is in understanding the role that a functional group plays in explaining phenotype heritability.

Conclusion

In conclusion, SLMM can accurately fit LMMs on large datasets comprising millions of rows and columns at a scale not previously possible. Through numerous simulations we show that SLMM provides an accurate estimate of variance components compared to the gold standard exact model. SLMM can be run in diverse computing contexts using a single or many computers, enabling analysis of extremely large datasets. SLMM is widely applicable, given the widespread use of LMMs in many genetic and genomic analysis methods, and is freely available as a Python library. In the future, we aim to implement improved versions of SLMM tailored towards quantitative and case/control GWAS, expand the model to support genetic and environmental interactions[52,53] as well as non-linear components[54,55].

Acknowledgements

This research has been conducted using the UK Biobank Resource under Application Number 51573. This research was enabled by support provided by SciNet (https://www.scinethpc.ca/), Compute Ontario (https://www.computeontario.ca/) and the Digital Research Alliance of Canada (alliancecan.ca).

References

1. Bycroft, C. *et al.* The UK Biobank resource with deep phenotyping and genomic data. *Nature* **562**, 203–209 (2018).
2. Our future health. *Our Future Health* https://ourfuturehealth.org.uk/.
3. Discover MVP data. https://www.mvp.va.gov/pwa/discover-mvp-data.
4. All of Us Research Program Investigators *et al.* The "All of Us" Research Program. *N. Engl. J. Med.* **381**, 668–676 (2019).
5. CanPath - Canadian partnership for tomorrow's health. *CanPath - Canadian Partnership for Tomorrow's Health* https://canpath.ca/ (2020).
6. Kurki, M. I. *et al.* FinnGen provides genetic insights from a well-phenotyped isolated population. *Nature* **613**, 508–518 (2023).
7. Nagai, A. *et al.* Overview of the BioBank Japan Project: Study design and profile. *J. Epidemiol.* **27**, S2–S8 (2017).
8. Feng, Y.-C. A. *et al.* Taiwan Biobank: A rich biomedical research database of the Taiwanese population. *Cell Genom.* **2**, 100197 (2022).
9. Orchard-Webb, D. The Shanghai Zhangjiang Biobank. *Biobanking.com* https://www.biobanking.com/the-shanghai-zhangjiang-biobank/ (2018).
10. 23andMe. DNA Genetic Testing For Health, Ancestry And More - 23andMe. https://www.23andme.com/.

11. Abdellaoui, A., Yengo, L., Verweij, K. J. H. & Visscher, P. M. 15 years of GWAS discovery: Realizing the promise. *Am. J. Hum. Genet.* **110**, 179–194 (2023).
12. Searle, S. R. *The Collected Works of Shayle R. Searle*. (Wiley-Blackwell, Hoboken, NJ, 2009).
13. Bates, D., Mächler, M., Bolker, B. & Walker, S. Fitting Linear Mixed-Effects Models using lme4. *arXiv [stat.CO]* (2014).
14. Yang, J. *et al.* Common SNPs explain a large proportion of the heritability for human height. *Nat. Genet.* **42**, 565–569 (2010).
15. Evans, L. M. *et al.* Comparison of methods that use whole genome data to estimate the heritability and genetic architecture of complex traits. *Nat. Genet.* **50**, 737–745 (2018).
16. Wainschtein, P. *et al.* Assessing the contribution of rare variants to complex trait heritability from whole-genome sequence data. *Nat. Genet.* **54**, 263–273 (2022).
17. Speed, D., Hemani, G., Johnson, M. R. & Balding, D. J. Improved heritability estimation from genome-wide SNPs. *Am. J. Hum. Genet.* **91**, 1011–1021 (2012).
18. Speed, D. *et al.* Reevaluation of SNP heritability in complex human traits. *Nat. Genet.* **49**, 986–992 (2017).
19. Zeng, J. *et al.* Signatures of negative selection in the genetic architecture of human complex traits. *Nat. Genet.* **50**, 746–753 (2018).
20. Evans, L. M. & Keller, M. C. Using partitioned heritability methods to explore genetic architecture. *Nature reviews. Genetics* vol. 19 185 (2018).
21. Loh, P.-R., Kichaev, G., Gazal, S., Schoech, A. P. & Price, A. L. Mixed-model association for biobank-scale datasets. *Nat. Genet.* **50**, 906–908 (2018).
22. Mbatchou, J. *et al.* Computationally efficient whole-genome regression for quantitative and binary traits. *Nat. Genet.* **53**, 1097–1103 (2021).
23. Jiang, L. *et al.* A resource-efficient tool for mixed model association analysis of large-scale data. *Nat. Genet.* **51**, 1749–1755 (2019).
24. Lippert, C. *et al.* FaST linear mixed models for genome-wide association studies. *Nat. Methods* **8**, 833–835 (2011).
25. Zhou, X. & Stephens, M. Efficient multivariate linear mixed model algorithms for genome-wide association studies. *Nat. Methods* **11**, 407–409 (2014).
26. Zhou, X. & Stephens, M. Genome-wide efficient mixed-model analysis for association studies. *Nat. Genet.* **44**, 821–824 (2012).
27. Yang, J., Lee, S. H., Goddard, M. E. & Visscher, P. M. GCTA: a tool for genome-wide complex trait analysis. *Am. J. Hum. Genet.* **88**, 76–82 (2011).
28. Pazokitoroudi, A. *et al.* Efficient variance components analysis across millions of genomes. *Nat. Commun.* **11**, 4020 (2020).
29. Loh, P.-R. *et al.* Contrasting genetic architectures of schizophrenia and other complex diseases using fast variance-components analysis. *Nat. Genet.* **47**, 1385–1392 (2015).
30. Wang, K. A. *et al.* Exact Gaussian Processes on a Million Data Points. *arXiv [cs.LG]* (2019).
31. Stein, M. L., Chen, J., Anitescu, M. & Others. Stochastic approximation of score functions for Gaussian processes. *Ann. Appl. Stat.* **7**, 1162–1191 (2013).
32. Gardner, J., Pleiss, G., Weinberger, K. Q., Bindel, D. & Wilson, A. G. GPyTorch: Blackbox Matrix-Matrix Gaussian Process Inference with GPU Acceleration. in *Advances in Neural Information Processing Systems 31* (eds. Bengio, S. et al.) 7576–7586 (Curran Associates, Inc., 2018).

33. Owen, A. B. Monte Carlo theory, methods and examples. https://artowen.su.domains/mc/ (2013).
34. Hutchinson, M. F. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Commun. Stat. Simul. Comput.* **19**, 433–450 (1990).
35. Frommer, A., Lund, K. & Szyld, D. B. Block Krylov subspace methods for functions of matrices. *Electron. Trans. Numer. Anal.* **27**, 100–126 (2018).
36. Xiang, Y.-F., Jing, Y.-F. & Huang, T.-Z. A New Projected Variant of the Deflated Block Conjugate Gradient Method. *J. Sci. Comput.* **80**, 1116–1138 (2019).
37. Martinsson, P.-G. & Tropp, J. Randomized Numerical Linear Algebra: Foundations & Algorithms. *arXiv [math.NA]* (2020).
38. Ubaru, S., Chen, J. & Saad, Y. Fast estimation of tr(f(A)) via stochastic lanczos quadrature. *SIAM J. Matrix Anal. Appl.* **38**, 1075–1099 (2017).
39. Schmidt, G., Nocedal, J. & Wright, S. *Numerical Optimization*. (Springer, New York, NY, 2008).
40. Wu, Y. & Sankararaman, S. A scalable estimator of SNP heritability for biobank-scale data. *Bioinformatics* **34**, i187–i194 (2018).
41. Liberty, E. & Zucker, S. W. The Mailman algorithm: A note on matrix–vector multiplication. *Inf. Process. Lett.* **109**, 179–182 (2009).
42. Blackford, L. S. *et al.* An updated set of basic linear algebra subprograms (BLAS). *ACM Trans. Math. Softw.* **28**, 135–151 (2002).
43. Searle, S. R. C. R. Henderson, the statistician; and his contributions to variance components estimation. *J. Dairy Sci.* **74**, 4035–4044 (1991).
44. Yang, J. *et al.* Genetic variance estimation with imputed variants finds negligible missing heritability for human height and body mass index. *Nat. Genet.* **47**, 1114–1120 (2015).
45. Speed, D., Holmes, J. & Balding, D. J. Evaluating and improving heritability models using summary statistics. *Nat. Genet.* **52**, 458–462 (2020).
46. Hou, K. *et al.* Accurate estimation of SNP-heritability from biobank-scale data irrespective of genetic architecture. *Nat. Genet.* **51**, 1244–1251 (2019).
47. Zhou, X. A unified framework for variance component estimation with summary statistics in genome-wide association studies. *Ann. Appl. Stat.* **11**, 2027–2051 (2017).
48. Gazal, S. *et al.* Linkage disequilibrium-dependent architecture of human complex traits shows action of negative selection. *Nat. Genet.* **49**, 1421–1427 (2017).
49. Finucane, H. K. *et al.* Partitioning heritability by functional annotation using genome-wide association summary statistics. *Nat. Genet.* **47**, 1228–1235 (2015).
50. Ning, Z., Pawitan, Y. & Shen, X. High-definition likelihood inference of genetic correlations across human complex traits. *Nat. Genet.* **52**, 859–864 (2020).
51. Gazal, S., Marquez-Luna, C., Finucane, H. K. & Price, A. L. Reconciling S-LDSC and LDAK functional enrichment estimates. *Nat. Genet.* **51**, 1202–1204 (2019).
52. Robinson, M. R. *et al.* Genotype-covariate interaction effects and the heritability of adult body mass index. *Nat. Genet.* **49**, 1174–1181 (2017).
53. Moore, R. *et al.* A linear mixed-model approach to study multivariate gene-environment interactions. *Nat. Genet.* **51**, 180–186 (2019).
54. Svensson, V., Teichmann, S. A. & Stegle, O. SpatialDE: identification of spatially variable genes. *Nat. Methods* **15**, 343–346 (2018).
55. Sun, S., Zhu, J. & Zhou, X. Statistical analysis of spatial expression patterns for spatially resolved transcriptomic studies. *Nat. Methods* **17**, 193–200 (2020).

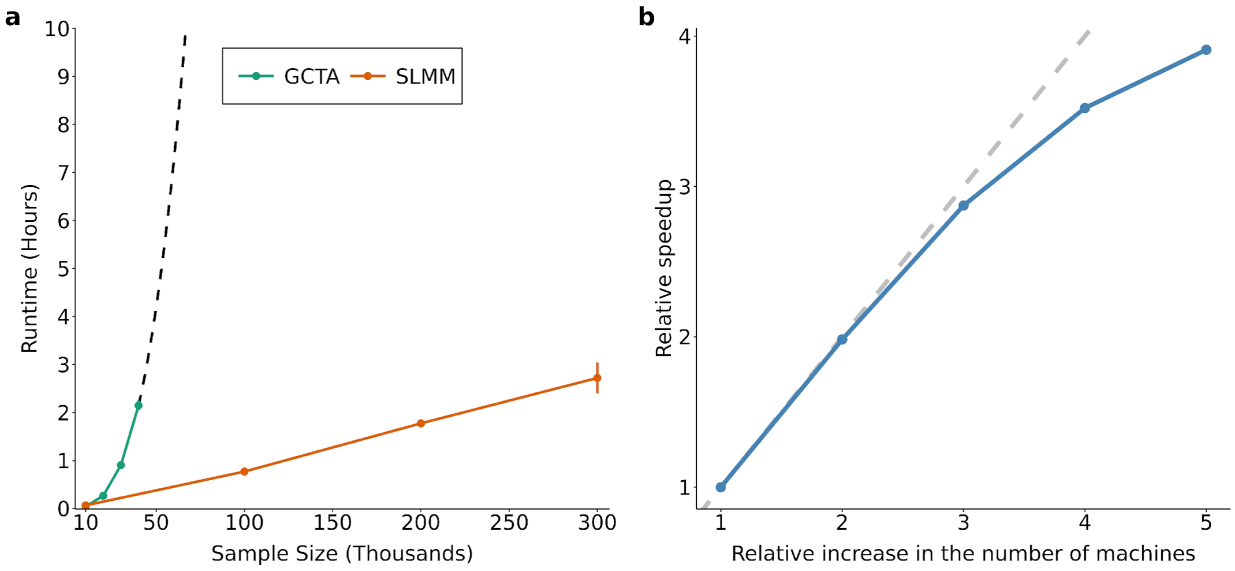


Figure 1. Runtime of SLMM. (a) Runtime comparison versus GCTA (numerically exact REML solver) based on total elapsed time in hours on a single compute node to fit a REML model of 22 components (representing each autosome in the human genome) on varying sample sizes on the UK Biobank data ("white British" samples). The total number of variants is approximately 415K SNPs. **(b)** Relative speedup with respect to increase in the number of compute nodes for running SLMM in a distributed manner. Each compute node (single and multi-node runs) contained 2 Intel Xeon Gold 6148 (2.40 GHz) CPUs (40 total physical cores).

Table 1: Performance of heritability models

		Average difference with the GCTA model across 23 phenotypes			
Heritability Model	**No. of Parameters**	**Heritability Difference (%)**	**Log-Likelihood**	**BIC**	**AIC**
LDAK	1	19.45	-945.93	1891.87	1891.87
LDAK-Thin	1	14.52	526.32	-1052.65	-1052.65
GREML LDMS-I	20	5.54	1359.87	-2478.13	-2681.73
GREML LDMS-R	20	-0.06	1131.00	-2020.40	-2224.00
Functional Model	47	9.26	3684.92	-6784.93	-7277.86

Log-likelihood: Log of the REML likelihood function (larger is better). **BIC:** Bayesian Information Criterion (smaller is better). **AIC:** Akaike Information Criterion (smaller is better). Values are averages over 23 UKB phenotypes analyzed in this paper. Note: the values are differences with the GCTA model and for the LDAK and LDAK-thin models, since the number of parameters are the same. This reflects twice the change in likelihood only and will be the same for BIC and AIC. **Functional Model:** Extension of the best model (GCTA LDMS-I) to include 27 additional components/kernels representing various functional SNP categories (See Methods, Supplementary Figures S11-14).

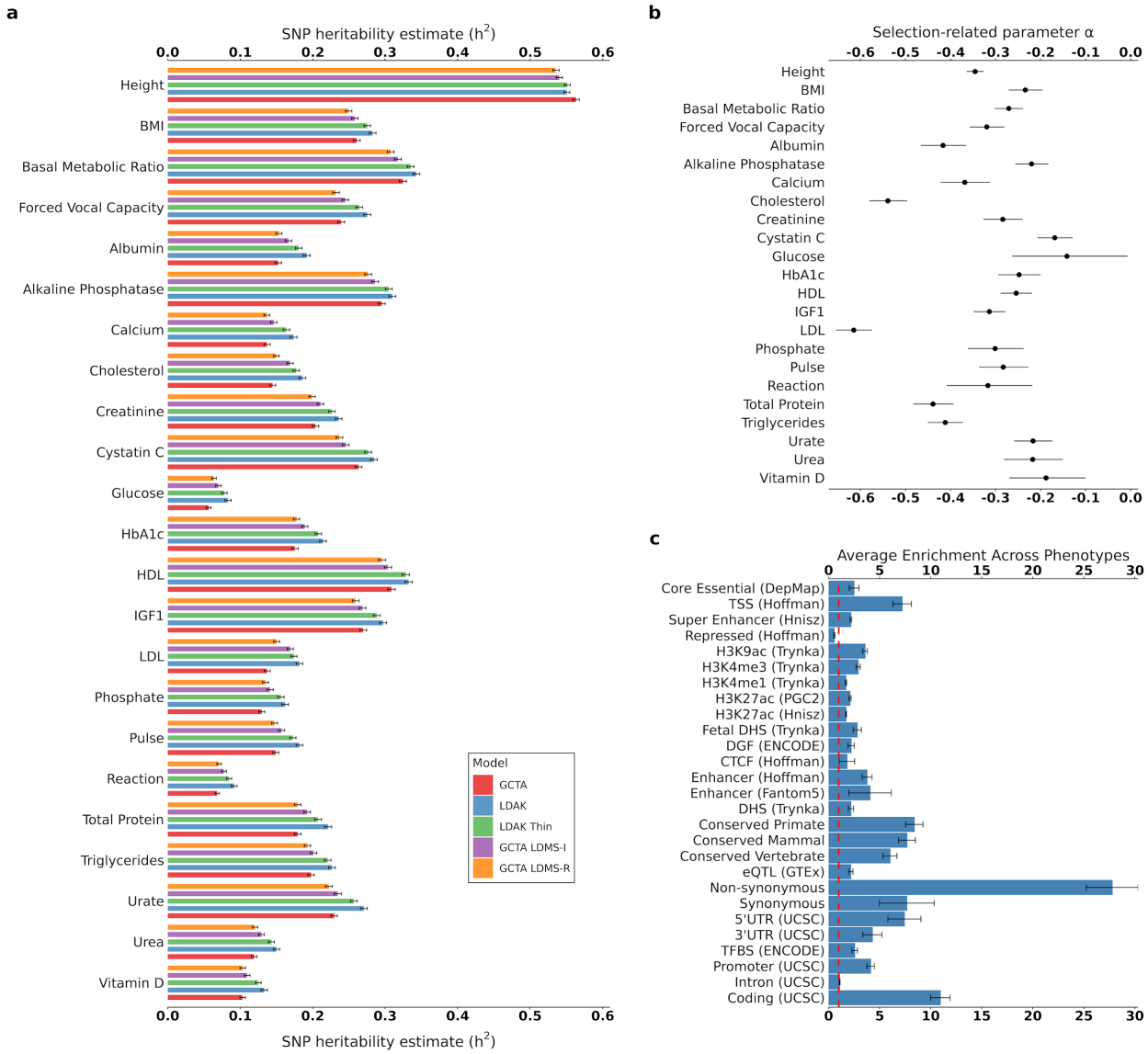


Figure 2: SLMM application results. (a) Estimates ($\pm 2 \times$ standard errors) of heritability estimates for 5 heritability models (GCTA, LDAK, LDAK-Thin, GREML LDMS-I, GREML LDMS-R) across 23 phenotypes from the UK Biobank. (b) Estimates ($\pm 2 \times$ standard errors) of selection-related parameter α using the LDAK-Thin model across the 23 phenotypes. More negative α means less frequent variants have larger effect sizes. (c) Estimates ($\pm 2 \times$ standard errors) of functional enrichment across 27 functional categories. The functional categories are mainly those of the Baseline LD model along with those in the Core Essential category (See Methods).

Methods

Linear Mixed Models

Consider the linear mixed model (LMM) as expressed below[1]

$$y = X\beta + Z_1 b_1 + \cdots + Z_K b_K + \epsilon = X\beta + Zb + \epsilon \quad (1)$$

where y is an $n \times 1$ vector of observations, X is an $n \times p$ design matrix for fixed effects β, $Z = [Z_1, \ldots, Z_K]$ is an $n \times q$ design matrix for random effects $b = [b_1^T, \ldots, b_K^T]^T$, Z_i is an $n \times q_i$ matrix, and ϵ is an $n \times 1$ vector of residual errors. Here K is the number of random effects/components. The superscript T means a transpose of a vector or matrix. We assume that the design matrix X is of full rank, satisfying conditions of estimability for the parameters. The random vectors b_i and ϵ are mutually uncorrelated and have a normal distribution, $b_i \sim N(0, \sigma_i^2 I_{q_i})$ and $\epsilon \sim N(0, \sigma^2 I_n)$, where I_n is an $n \times n$ identity matrix.

The unknown parameter $\sigma_k^2, k = 1, \ldots, K$, and σ^2 are called variance components. Let $\sigma_0^2 = \sigma$, $Z_0 = I_n$, and $V_i = Z_i Z_i^T$. Define $\theta_k = \sigma_k^2$ and $\theta = [\theta_0, \theta_1, \ldots, \theta_k]^T$. The log-likelihood function is given as

$$l(\beta, \theta) = log\, f(y|\beta, \theta) = -\frac{1}{2} n\, log\, 2\pi - \frac{1}{2} log\, det(V_\theta) - \frac{1}{2}(y - X\beta)^T V_\theta^{-1}(y - X\beta) \quad (2)$$

where $V_\theta = Cov(y) = \Sigma_{i=0}^K \sigma_i V_i$. Refer to the Supplementary Notes Section 1 for more information.

Fitting LMM using REML: Patterson and Thompson[2] proposed a modified maximum likelihood procedure which partitions the data into two mutually uncorrelated parts, one being free of the fixed effects used for estimating the variance components, called restricted/residual maximum likelihood (REML) estimators (Refer to Supplementary Notes Section 1.2 for detailed more information). Let M be an $n \times (n - p)$ full column rank matrix such that $M^T X = 0$, and $L = V_\theta^{-1} X$, an $n \times p$ full column rank matrix. Then the data y can be partitioned into two parts: $z = M^T y$ and $u = L^T y$. The data z and u are uncorrelated since $Cov(z, u) = M^T V_\theta L = M^T X = 0$. The transformed data, z, which does not contain the fixed effects β, has the log-likelihood function:

$$l_R(\theta) = -\frac{1}{2}(n - p) log 2\pi - \frac{1}{2} logdet(M^T V_\theta M) - \frac{1}{2} y^T M (M^T V_\theta M)^{-1} M^T y. \quad (3)$$

The REML estimates of θ are obtained by $\theta = arg\, l_R(\theta)$. Based on the second part of the data, u, the maximum likelihood estimation (MLE) of β is given by

$$\hat{\beta} = \left(X^T V_\theta^{-1} X\right)^{-1} X^T V_\theta^{-1} y \quad (4)$$

Under regularity conditions, the REML estimators are consistent and asymptotically normal with asymptotic covariance matrix equal to the inverse of Fisher information matrix, that is, asymptotically[3]

$$\sqrt{n}\left(\hat{\theta}-\theta\right) \sim N(0, I(\theta)^{-1}) \tag{5}$$

where $I(\theta) = -E\left(\frac{\partial^2 l_R}{\partial\theta_i \partial\theta_j}\right) = \left\{\frac{1}{2} tr\left(R_\theta V_i R_\theta V_j\right)\right\}_{0\le i,j\le K}$ is the Fisher information matrix, and

$$R_\theta = V_\theta^{-1} - V_\theta^{-1} X\left(X^T V_\theta^{-1} X\right)^{-1} X^T V_\theta^{-1} = M(M^T V_\theta M)^{-1} M^T. \tag{6}$$

Moreover, for a differentiable function $g(\theta)$, $g\left(\hat{\theta}\right)$ is the MLE of $g(\theta)$, and asymptotically

$$\sqrt{n}\left(g\left(\hat{\theta}\right) - g(\theta)\right) \sim N(0, \nabla g^T I(\theta)^{-1} \nabla g) \tag{7}$$

where $\nabla g = \frac{\partial g(\theta)}{\partial\theta}$ is the first partial derivatives of g. Standard errors are calculated using the above formulation. Refer to Supplementary Notes Section 1.2 for more information.

Bottlenecks in REML computations: With the variance components estimated, the fixed effects are given by (4). Finding the variance components is a difficult numerical problem. Maximizing the log likelihood function with respect to the variance components is a nonlinear optimization problem. Various iterative methods based on the log likelihood have been proposed to compute the REML, including gradient methods based on derivatives. The gradient methods are represented by the iteration equation[1]

$$\theta^{(m+1)} = \theta^{(m)} + Q\left(\theta^{(m)}\right)\frac{\partial l\left(\theta^{(m)}\right)}{\partial\theta} \tag{8}$$

where $\partial l(\theta)/\partial\theta$ is the gradient of the log likelihood function, and $Q(\theta)$ is a modifier matrix of the gradient direction. Let $H(\theta)$ and $I(\theta)$ be the Hessian matrix and information matrix of the log likelihood function with respect to θ. Let $I_A(\theta) = \left[I(\theta) + \left(-H(\theta)\right)\right]/2$ be the average information matrix. The modifier matrix can be specified by 1. Newton–Raphson: $Q(\theta) = -H(\theta)^{-1}$; 2. Fisher scoring: $Q(\theta) = I(\theta)^{-1}$; 3. Average information: $Q(\theta) = I_A(\theta)^{-1}$. The elements of the gradient, $H(\theta)$, $I(\theta)$ and $I_A(\theta)$, respectively, given as follows:

$$\frac{\partial l_R}{\partial\theta_i} = -\frac{1}{2}[tr(R_\theta V_i) - y^T R_\theta V_i R_\theta y],$$

$$H_{ij} = \frac{\partial^2 l_R}{\partial\theta_i \partial\theta_j} = \frac{1}{2} tr\left(R_\theta V_i R_\theta V_j\right) - y^T R_\theta V_i R_\theta V_j R_\theta y,$$

$$I_{ij} = -E\left(H_{ij}\right) = \frac{1}{2} tr\left(R_\theta V_i R_\theta V_j\right),$$

$$I_{A,ij} = \frac{1}{2}\left(I_{ij} - H_{ij}\right) = \frac{1}{2} y^T R_\theta V_i R_\theta V_j R_\theta y.$$

The difficulties in REML computation arise from following factors:

1. matrix inversion: V_θ^{-1}

2. matrix-matrix products in the traces: $\mathrm{tr}(R_\theta V_i)$ and $\mathrm{tr}\left(R_\theta V_i R_\theta V_j\right)$
3. matrix log-determinants: $\log|V_\theta|$
4. non-linear optimization (needs a good starting point, prone to divergence)
5. Forming V_θ requires matrix multiplication as $V_\theta = \Sigma_{i=0}^{K}\sigma_i V_i = \Sigma_{i=0}^{K}\sigma_i Z_i Z_i^T$
6. Steep memory requirement for large Z_i matrices.

Scalable LMM (SLMM)

We developed a scalable algorithm for LMM estimation using REML that addresses the difficulties listed above using several tools and techniques described below. Refer to Supplementary Notes Section 2 for detailed information on each topic.

Trace estimation: we avoided matrix-matrix product by using stochastic trace estimation[4] (Supplementary Notes Section 2.1). For the average information, fitting REML requires computing $\mathrm{tr}(R_\theta V_i)$ which can be estimated with stochastic trace estimation:

$$tr(R_\theta V_i) \approx \frac{1}{L}\sum_{l=1}^{L}\left(u_l^T R_\theta\right)(V_i u_l),$$

$$R_\theta u_l = M(M^T V_\theta M)^{-1} M^T u_l,$$

where u_l, $l = 1, \ldots, L$, are L independent random vector (random probes) with mean zero and covariance I_n from a probability distribution such as Rademacher. The trace estimates need to compute the inverse matrix-vector products: $(M^T V_\theta M)^{-1} v$, where $v_l = M u_l$. These trace estimates could suffer from large variance requiring large number of probes (L).

Variance reduction: The variance of trace estimate can be reduced by variance reduction techniques, such as control variates[5] (Supplementary Notes Section 2.2) resulting in significant speedups (fewer probes) in computations as well as increase in estimation accuracy. Let $f(u) = u^T A u$. We define a control variate as the random variable $g(u) = u^T B u$ that has a known mean, $E(g) = tr[BCov(u)] = tr(B)$. For any given b,

$$f_b(u) = f(u) - b[g(u) - E(g)]$$

can be used for an estimate of the trace $tr(A)$ since $E(f_b) = tr(A)$. The variance is

$$Var(f_b) = Var(f) - 2bCov(f, g) + b^2 Var(g).$$

The optimal coefficient b^* is given by

$$b^* = \underset{b}{argmin} Var(f_b) = \frac{Cov(f,g)}{Var(g)}.$$

Since

$$Var(f_{b^*}) = Var(f)[1 - \frac{Cov^2(f,g)}{Var(f)Var(g)}] = Var(f)[1 - cor^2(f,g)],$$

the variance of the trace estimate with a control variate is reduced.

Linear systems and matrix inversion: We computed the inverse matrix-vector product above along with the one required for computing the quadratic form in both gradient and log-likelihood (i.e. $y^T R_\theta y$) without explicitly forming and inverting the matrix V_θ using efficient block Krylov solvers[6] (Supplementary Notes Section 2.5) namely block Lanczos[6–8] and deflated preconditioned block conjugate gradient[9] which allow us to simultaneously compute the inverse matrix-vector products in an efficient manner:

$$(M^T V_\theta M)^{-1} B = (M^T V_\theta M)^{-1}[My, U]$$

where $U = [Mu_1, \dots, Mu_L]$. The block Lanczos algorithm can solve both indefinite and positive (semi)-definite systems. For the latter, the block Lanczos algorithm gives the block conjugate gradient algorithm. Refer to supplementary information for more information.

The log-likelihood function: Computing the log-likelihood function by (3) requires the computation of log-determinant, $\mathsf{logdet}\,(M^T V_\theta M)$, and inverse matrix-vector product, $(M^T V_\theta M)^{-1} M^T y$. The inverse matrix-vector can be computed using the efficient solvers mentioned above and is already calculated for the computation of trace terms. As for the log-determinant, we used an accurate and efficient method that estimates this quantity using a combination of stochastic trace estimation, Gaussian quadrature and the Lanczos algorithm[6,10,11]. Refer to Supplementary Notes Section 2.6 for more information.

Matrix-free operations: The solutions mentioned above do not require explicit formation of matrix V_θ, therefore allowing us to avoid the expensive matrix-matrix products for forming the covariance matrix. Refer to Supplementary Notes Section 2.4 for more information.

Optimization and pre-training: We used the trust-region algorithm[12] to achieve global convergence and used a pre-training strategy to find a good starting point by pooling estimates from the subset of data. Refer to Supplementary Notes Sections 2.7 and 2.8 for more information. Using an optimization strategy for avoiding saddle points and achieving global convergence is essential otherwise the model may face many divergences[12] as is the case with the GCTA software which we observed fails to converge in a variety of simulations.

Mailman algorithm for efficient genotype-vector products: We used the mailman algorithm[13,14] which is an efficient algorithm for sub-linear matrix-vectors involving matrices with limited number of possible values as is the case with the genotype data (values: 0,1,2) (Supplementary Notes Section 2.3).

Computational complexity: Overall, the computational complexity of the SLMM estimation is $O\left(\frac{mnL\sqrt{\kappa_L}}{max\,((m),(n))}\right)$ where m is the number of variants (columns) in the genotype matrix (random effects), n is the number of samples (rows), L is the number of random vectors (probes) for stochastic trace estimation and $\kappa_L = \lambda_n/\lambda_L$ where λ_n and λ_1 represent the largest and the smallest eigenvalues of the covariance matrix and λ_L is the L-th smallest eigenvalue of the covariance matrix.

Efficient in-core, out-of-core and distributed computation: SLMM is available as a versatile Python library that supports in-core, out-of-core, and distributed training of models in a high-performance computing (HPC) cluster or cloud. The implementation makes use of ahead-of-time (AOT) and just-in-time (JIT) compilation (through Pythran[15] and Numba[16] libraries respectively) for achieving high performance without sacrificing readability and ease of access and modification for the scientific community. As for the multi-machine implementation, SLMM includes a custom approach for efficient cross-compute node deployment and communication to efficiently set up a distributed system of worker nodes each handling a batch of genotype data, and a manager thread that handles deployment, efficient communications (function and argument transfer to worker nodes and gathering of results) and monitoring. From a user's perspective, flexibility in the run scenario is available with minimal to no modifications of job parameters, meaning that switching between in-core and out-of-core operation is a simple Boolean parameter, and the difference between single-machine in-core run and a distributed one is if the user passes a list of host names (another parameter of the SLMM function call). We have simplified this even further for Slurm compute clusters (https://slurm.schedmd.com) such that no host list is needed and the same in-core script that runs on a consumer device if run on a multi-node job will automatically detect the nodes and run in a fully distributed manner.

Dataset, Simulation and Benchmarking

Dataset: Real genotype data used in this study was that of the UK Biobank accessed under application 51573. For details related to phenotypes and fixed effects used in our analysis refer to the Supplementary Notes Sections 3.3 and 3.4.

Preprocessing: For the real data analysis, we used the full imputed genotype data provided by the UK Biobank and downloaded sample indices from the UK Biobank GWAS repository of the Nealelab v3 (https://github.com/Nealelab/UK_Biobank_GWAS). We then filtered the variants to keep those with

MAF ≥ 0.01 and information score ≥ 0.99. This left us with a dataset of 357,600 samples and 4,996,895 imputed variants which we converted to hard calls using standard procedures in Plink v2.0.

Simulations: To compare the statistical accuracy and speed of SLMM with the state-of-the-art, we performed benchmarking and simulation using real genotypes where enough samples were available in the UK Biobank data (e.g., 10K and 300K simulations) and used a genotype simulation strategy to generate large datasets (e.g., 5M samples). **10K data:** Real 10,000 random subset of the UK biobank white British samples with approximately 415,000 variants (genotyped SNPs). Analysis is repeated 1000 times unless stated otherwise. The number of components is set to 20 representing the first 20 autosomes unless stated otherwise. **300K Data:** Real 300,000 random subset of the UK biobank white British samples with approximately 146,000 variants (genotyped SNPs) representing the first 5 autosomes. The number of components is set to 5, one for each chromosome. **5M Data:** To evaluate the performance of SLMM on extremely large datasets involving millions of samples we used genotype simulation. Following established procedures[17–19], we used haplotype resampling-based strategies to generate a dataset of 5 million individuals of varying degrees of relatedness. More specifically, we varied the number of ancestors for generating a simulated individual such that the dataset contains individuals with varying degrees of relatedness. At a high level, in the resampling-based approach one selects a number of individuals as ancestors and having defined haplotypes or naively splitting the genome to a number of blocks, a simulated genotype can be generated by randomly choosing each block (haplotype) from one of the ancestors. The parameters that affect the relatedness between individuals in the generated dataset are the number of ancestors and how many samples are generated from the same group of ancestors. For example, repeated sample generation from two ancestors would lead to generating family members (siblings) while a larger number, such as randomly selecting 10 individuals from the pool, and generating a sample would result in approximately unrelated individuals[17]. We used the genotyped SNPs in this simulation and after removing rare variants we ended up with a simulated genotype data of 5 million samples and 302,379 variants which was used in distributed runs.

Runtime environment: All the runs and comparisons were performed using the Niagara cluster of the Scinet advanced research computing facility at the University of Toronto (https://www.scinethpc.ca/). Each node in the cluster contained 2 Intel Xeon Gold 6148 CPUs, each with 20 physical cores (40 cores in total) and 180 GiB of memory. Data was stored on SSD storage to achieve high read/write performance.

Analysis

Heritability and enrichment estimation: Consider the GCTA model with K components described as follows:

$$y = X\beta + G\gamma + \epsilon = X\beta + G_1\gamma_1 + \cdots + G_K\gamma_K + \epsilon \tag{20}$$

where $G = [G_1, \ldots, G_K]$, and $G_k = \left[g_{k1}, \ldots, g_{k,m_k}\right]$ is an $n \times m_k$ matrix of genotypes in the group k. K is the number of groups. The vector of random effects, $\gamma = [\gamma_1^T, \ldots, \gamma_K^T]^T$, follows a multivariate normal distribution with $E(\gamma) = 0$ and $Cov(\gamma) = diag\left(\sigma_1^2 I_{m_1}, \ldots, \sigma_K^2 I_{m_K}\right)$. We obtain the heritabilities for all "causal" SNPs (h_G), a subset S of causal SNPs (h_S), and enrichment (r_S) of the subset in the GCTA model as follows:

$$h_G = \frac{\sum_{k=1}^{K} \sigma_k^2 w_k}{\sigma^2 + \sum_{k=1}^{K} \sigma_k^2 w_k}, \tag{21}$$

$$h_S = \frac{\sum_{k=1}^{K} \sigma_k^2 u_k}{\sigma^2 + \sum_{k=1}^{K} \sigma_k^2 w_k}, \tag{22}$$

$$r_S = \frac{m \sum_{k=1}^{K} \sigma_k^2 u_k}{m_S \sum_{k=1}^{K} \sigma_k^2 w_k}, \tag{23}$$

where

$$w_k = \sum_{j=1}^{m_k} v_{kj} = \frac{1}{n-1} \sum_{j=1}^{m_k} g_{kj}^T P_n g_{kj} = \frac{1}{n-1} tr\left(G_k^T P_n G_k\right),$$

$$u_k = \sum_{\mathrm{SNP}_{kj} \in S} v_{kj} = \frac{1}{n-1} \sum_{\mathrm{SNP}_{kj} \in S} g_{kj}^T P_n g_{kj}.$$

$v_{kj} = \frac{1}{n-1} g_{kj}^T P_n g_{kj}$ is the sample variance of the genotypes g_{kj}, $P_n = I_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T$, and $E\left(v_{kj}\right) = 2f_{kj}\left(1 - f_{kj}\right)$. Refer to Sections 4.1 to 4.3 in the Supplementary Notes for detailed information on the derivation of heritability and related quantities.

Functional Standards: In functional enrichment analysis, we expanded the best model GREML LDMS-I to include 27 additional components each representing a group of variants assigned to the functional group; 26 of these functional modules are those from the Baseline LD model provided in the S-LDSC software[20]. We defined the last component, referred to as core essential, to include variants that locate within core essential genes as defined in DepMap[21].

Selection related parameter: In the LDAK-thin[22] model we have $\gamma \sim N(0, \sigma^2 D)$ where D is an $m \times m$ diagonal matrix with $D_{ii} = \delta_i w_i / m$ where $w_i = [f_i(1 - f_i)]^\alpha$, f_i is the MAF of the ith SNP and δ_i specifies whether ith SNP remains in the model after the thinning process. Studies[22–24] have shown that the α parameter specifies the relationship between MAF and heritability and has been used to measure selection

(negative estimates indicate less common SNPs tend to have larger effect sizes and vice versa). We used SLMM to estimate α using profile likelihood by choosing a set of possible values for α and fitting the model using each selected value[22]. Profile likelihood is a standard procedure in statistical literature for inferring a parameter and its sampling distribution by profiling out the rest of the parameters (as nuisance parameters)[25–27]. The primary reason for doing this is that allowing α to be a free parameter in the model would make the covariance non-linear, which is not currently supported in SLMM.

References

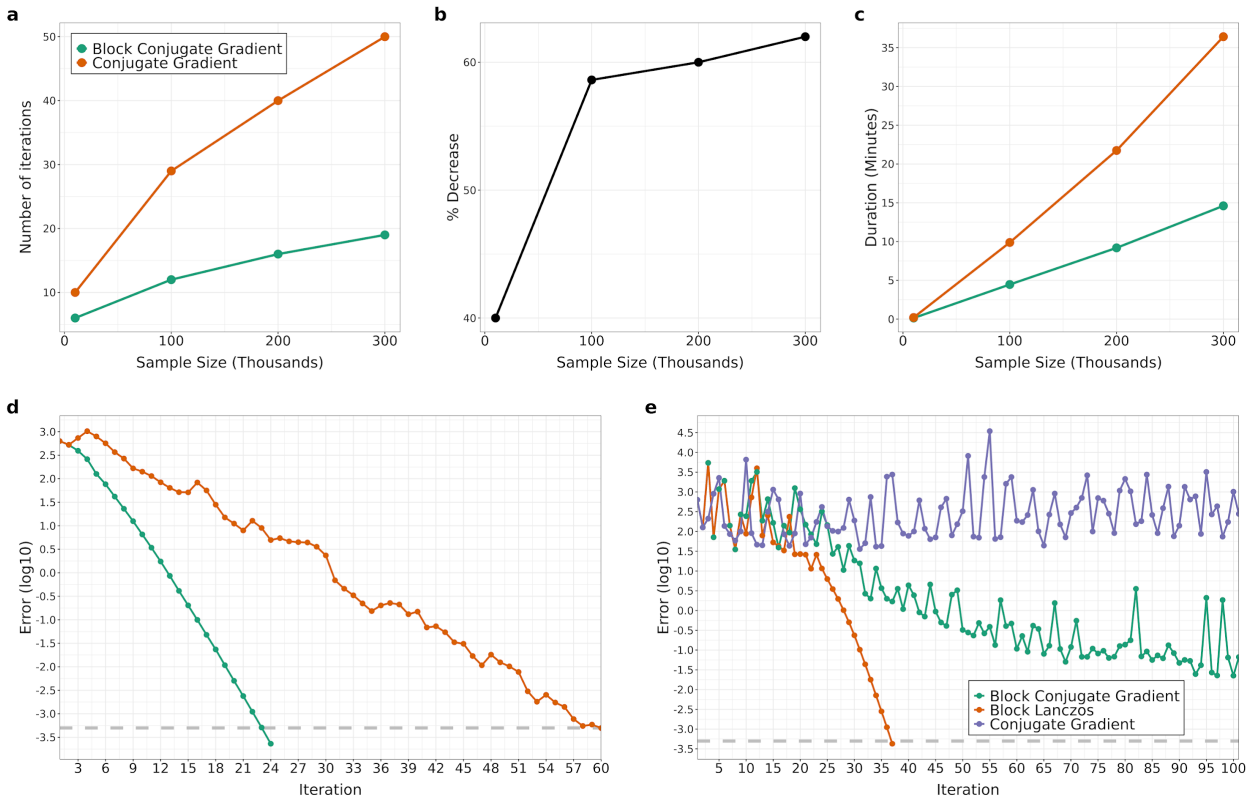
1. Searle, S. R., Casella, G. & McCulloch, C. E. *Variance Components*. (John Wiley & Sons, 2009).
2. Patterson, H. D. & Thompson, R. Recovery of inter-block information when block sizes are unequal. *Biometrika* **58**, 545–554 (1971).
3. Berger, R. & Casella, G. *Statistical Inference*. (Duxbury Press, Florence, AL, 2001).
4. Hutchinson, M. F. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Commun. Stat. Simul. Comput.* **19**, 433–450 (1990).
5. Owen, A. B. Monte Carlo theory, methods and examples. https://artowen.su.domains/mc/ (2013).
6. Frommer, A., Lund, K. & Szyld, D. B. Block Krylov subspace methods for functions of matrices. *Electron. Trans. Numer. Anal.* **27**, 100–126 (2018).
7. Cullum, J. & Donath, W. A block Lanczos algorithm for computing the q algebraically largest eigenvalues and a corresponding eigenspace of large, sparse, real symmetric matrices. *1974 IEEE Conference on Decision and Control including the 13th Symposium on Adaptive Processes* Preprint at https://doi.org/10.1109/cdc.1974.270490 (1974).
8. Golub, G. H. & Underwood, R. The block lanczos method for computing eigenvalues. in *Mathematical Software* 361–377 (Elsevier, 1977).
9. Xiang, Y.-F., Jing, Y.-F. & Huang, T.-Z. A New Projected Variant of the Deflated Block Conjugate Gradient Method. *J. Sci. Comput.* **80**, 1116–1138 (2019).
10. Ubaru, S., Chen, J. & Saad, Y. Fast estimation of tr(f(A)) via stochastic lanczos quadrature. *SIAM J. Matrix Anal. Appl.* **38**, 1075–1099 (2017).
11. Martinsson, P.-G. & Tropp, J. Randomized Numerical Linear Algebra: Foundations & Algorithms. *arXiv [math.NA]* (2020).
12. Schmidt, G., Nocedal, J. & Wright, S. *Numerical Optimization*. (Springer, New York, NY, 2008).
13. Liberty, E. & Zucker, S. W. The Mailman algorithm: A note on matrix–vector multiplication. *Inf. Process. Lett.* **109**, 179–182 (2009).
14. Wu, Y. & Sankararaman, S. A scalable estimator of SNP heritability for biobank-scale data. *Bioinformatics* **34**, i187–i194 (2018).
15. Guelton, S. *et al.* Pythran: enabling static optimization of scientific Python programs. *Comput. Sci. Discov.* **8**, 014001 (2015).
16. Lam, S. K., Pitrou, A. & Seibert, S. Numba: a LLVM-based Python JIT compiler. in *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC* 1–6 (Association for Computing Machinery, New York, NY, USA, 2015).
17. Loh, P.-R. *et al.* Efficient Bayesian mixed-model analysis increases association power in large cohorts. *Nat. Genet.* **47**, 284–290 (2015).

18. Jiang, L. *et al.* A resource-efficient tool for mixed model association analysis of large-scale data. *Nat. Genet.* **51**, 1749–1755 (2019).
19. Casale, F. P., Rakitsch, B., Lippert, C. & Stegle, O. Efficient set tests for the genetic analysis of correlated traits. *Nat. Methods* **12**, 755–758 (2015).
20. Gazal, S. *et al.* Linkage disequilibrium-dependent architecture of human complex traits shows action of negative selection. *Nat. Genet.* **49**, 1421–1427 (2017).
21. Ghandi, M. *et al.* Next-generation characterization of the Cancer Cell Line Encyclopedia. *Nature* **569**, 503–508 (2019).
22. Speed, D., Holmes, J. & Balding, D. J. Evaluating and improving heritability models using summary statistics. *Nat. Genet.* **52**, 458–462 (2020).
23. Speed, D. *et al.* Reevaluation of SNP heritability in complex human traits. *Nat. Genet.* **49**, 986–992 (2017).
24. Zeng, J. *et al.* Signatures of negative selection in the genetic architecture of human complex traits. *Nat. Genet.* **50**, 746–753 (2018).
25. Murphy, S. A. & Van Der Vaart, A. W. On profile likelihood. *J. Am. Stat. Assoc.* **95**, 449–465 (2000).
26. Cole, S. R., Chu, H. & Greenland, S. Maximum likelihood, profile likelihood, and penalized likelihood: a primer. *Am. J. Epidemiol.* **179**, 252–260 (2014).
27. Bates, D., Mächler, M., Bolker, B. & Walker, S. Fitting Linear Mixed-Effects Models using lme4. (2014).

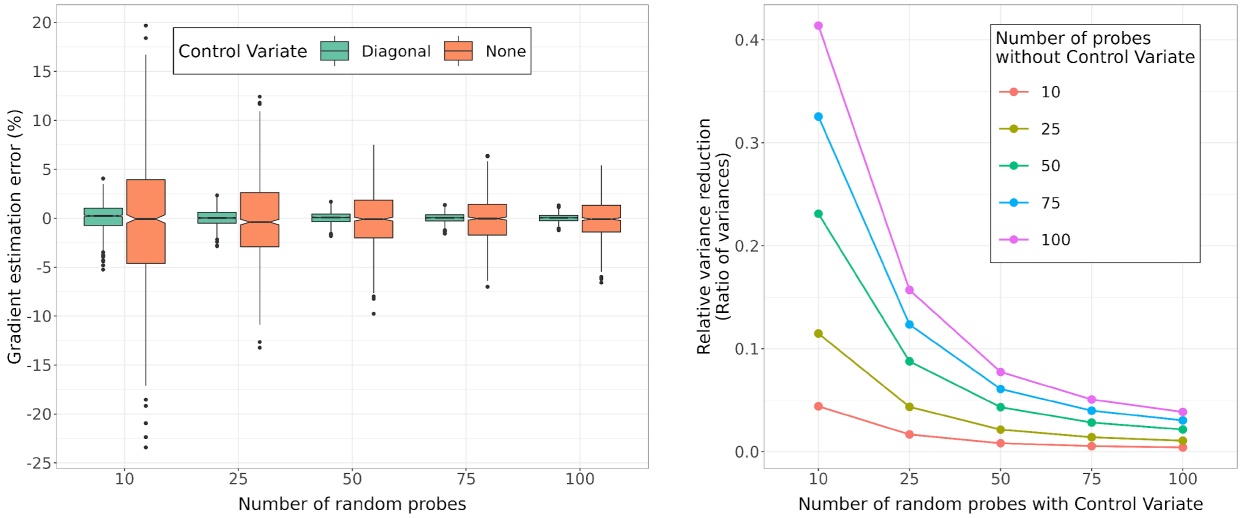
Supplementary information for

Genotype to phenotype analysis on the scale of millions of human genomes

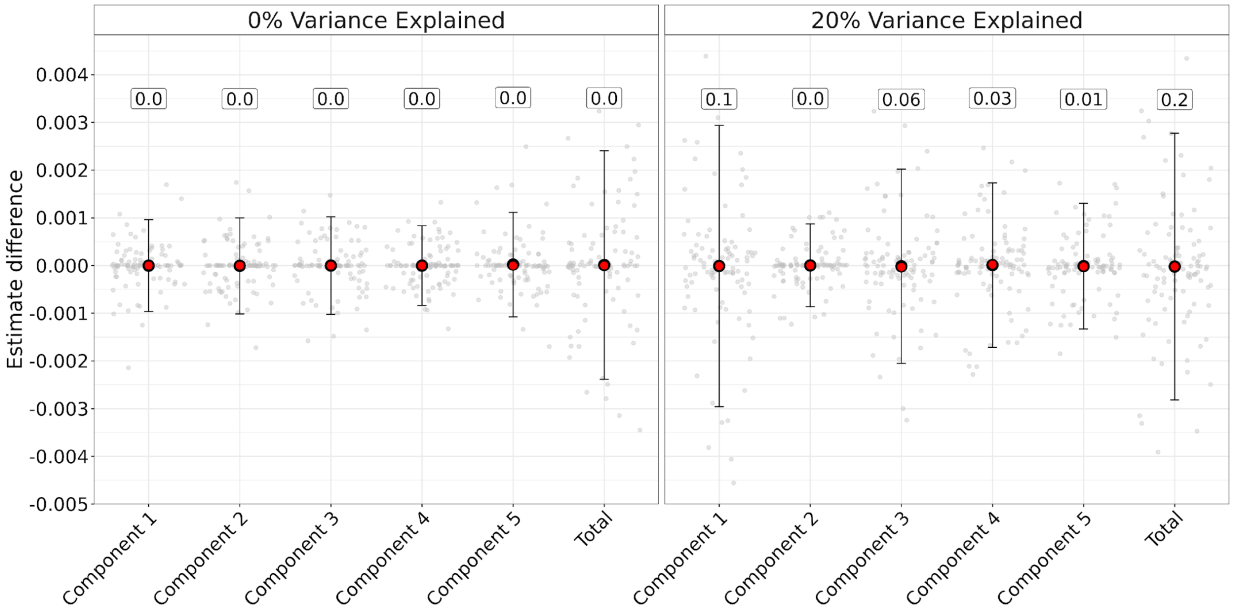
Hamed Heydari, Changjiang Xu, Charles Boone, Gary D. Bader



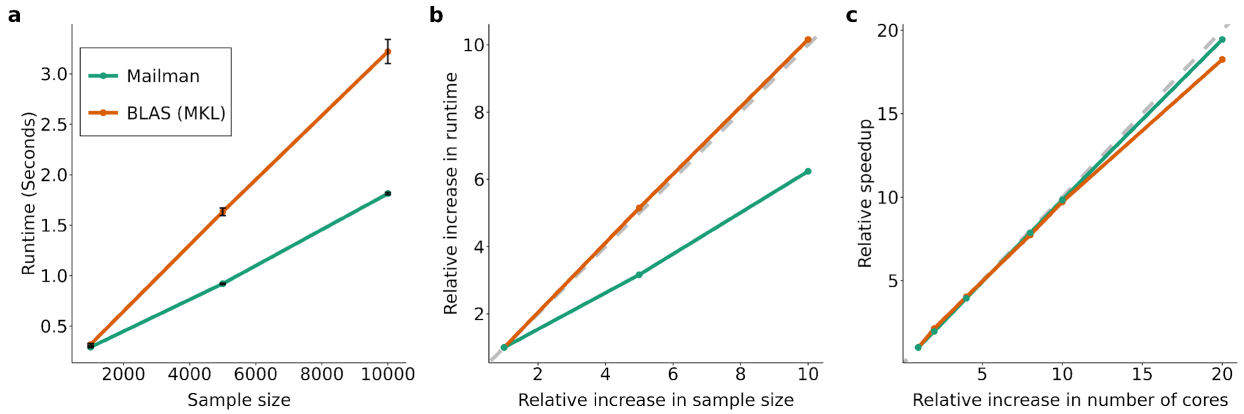
Supplementary Figure 1. Efficient solvers used in SLMM are faster and more convergent than the traditional conjugate gradient (CG) solver. Data for all analyses is the 300K subset of the UK biobank data (white British, see methods). The covariance matrix in the linear system is made up of a single linear kernel (the genetic relationship matrix, GRM) and the residual/noise component. Genetic variance component is set to 0.25. The solve is performed against a matrix containing 10 columns (10 linear systems). CG solves the systems independently, performing only the multiplications in block format (multiplying the GRM with all vectors together to increase efficiency, using BLAS level 3 matrix-matrix multiplication). Block solvers perform both the solves and multiplication in block format (the difference with CG is not in the BLAS call, but in the rest of the algorithm). (a) Number of iterations to solve linear systems to tolerance of 0.0005. (b) Percent decrease in number of iterations in the block solver versus CG without block solver. (c) Total time for solving the systems. (d) Example progress of solvers to achieve convergence (400K subset of the UK Biobank, approx. 415K array SNPs, parameters are the same as above). (e) Performance of solvers on indefinite systems (GRM made of 400K subset of the UK Biobank, approx. 415K array SNPs, genetic linear kernel parameter set to -0.05 and residual to 1.05).



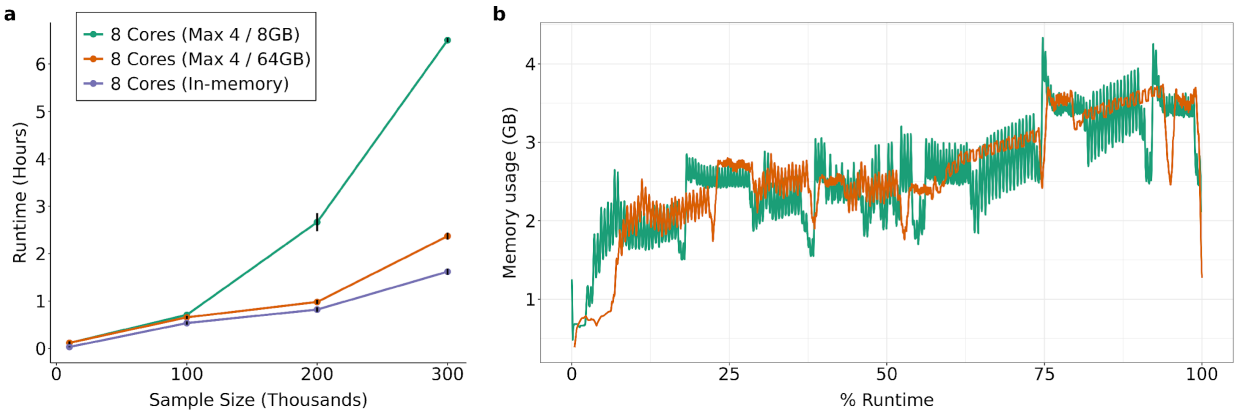
Supplementary Figure 2. Variance reduction increases the accuracy of gradient estimation Variance reduction increases accuracy by decreasing the required number of random probes needed for precise estimation of the gradient. Left plot shows the gradient estimation error for a model trained on real data (10,000-sample subset of the UK Biobank with approximately 400,000 SNPs). Estimation is repeated 1,000 times by randomly sampling the desired number of probes and performing gradient evaluation with and without control variate. Right plot compares the relative variance of trace estimation (the variance of estimate with control variate divided by that of the estimate without control variate) across different random probe numbers. For example, it appears that trace estimation using control variate with 10 random probes is more accurate than with 100 probes and no control variate (variance of the former is approximately 40% of the latter).



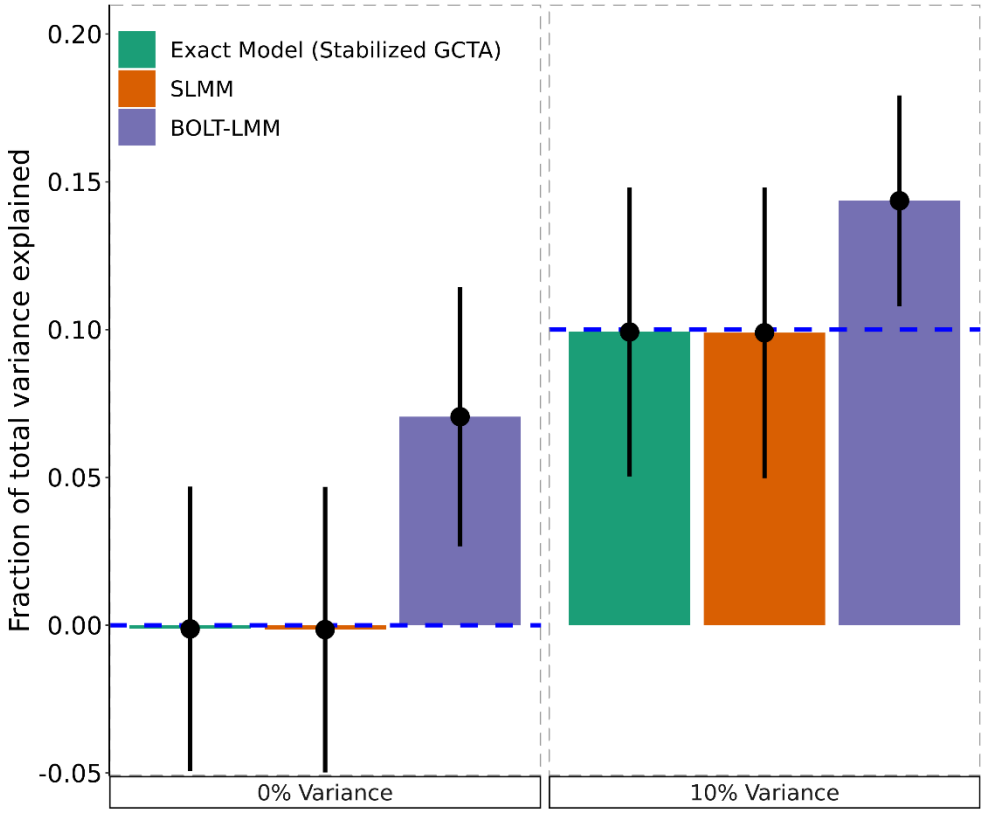
Supplementary Figure 3. Reducing the number of random probes in trace estimation with variance reduction on model parameter estimates with a large sample size does not greatly affect estimates, but decreases the runtime. Plots illustrate the difference between estimates on large scale simulations (N=300,000 genotype data from the UK Biobank) with 100 or 25 probes on a model with 5 components (kernels) and two phenotype simulation scenarios (0% and 20% variance explained by all components). The number of simulations is 100 per scenario. Black points represent the mean difference (±standard deviation). Red points are median. Labels (numbers in boxes above each distribution) on the plot show the true value of parameters used to simulate the phenotype/response. Reducing the number of probes from 100 to 25 improves performance of SLMM by four times.



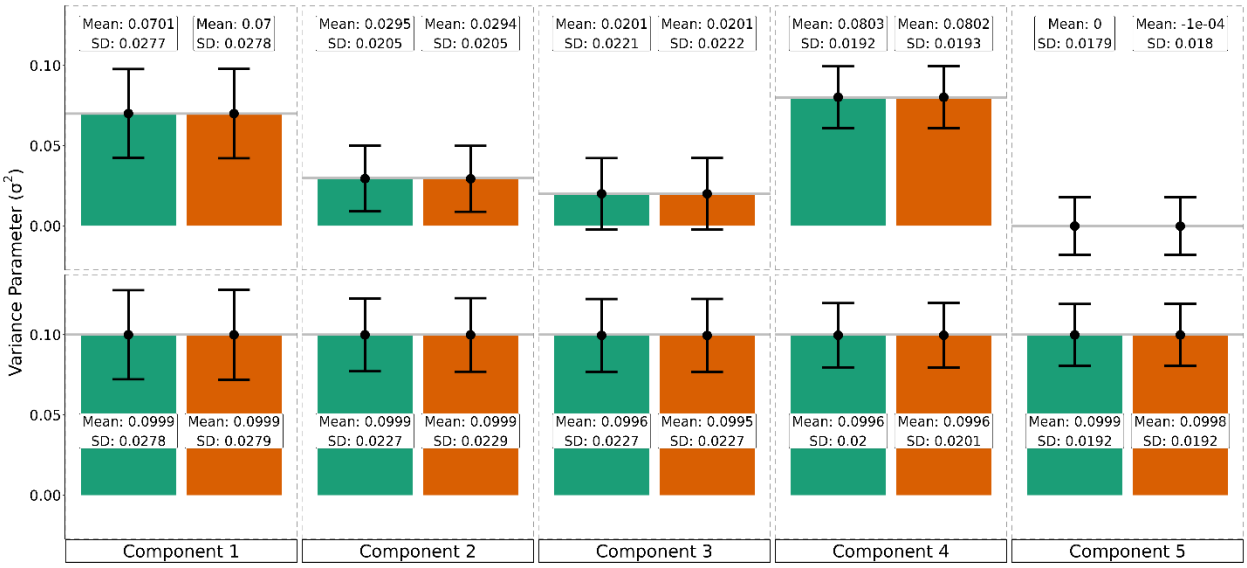
Supplementary Figure 4. Efficient implementation of the mailman algorithm achieves sublinear complexity and surpasses the performance of the highly efficient BLAS library (Intel's MKL) by using the reduced alphabet of genotype data. Data is a 10K subset of the UK Biobank data, approx. 415K array SNPs. Timings are for multiplying the genotype matrix (10K by 415K) with a matrix of (415K by 10). (a) Total elapsed time ($\pm\mathbf{2}\times$ standard deviations) (b) Relative increase in runtime with respect to increase in sample size (c) Relative speedup with respect to increase in number of threads (here, physical cores).



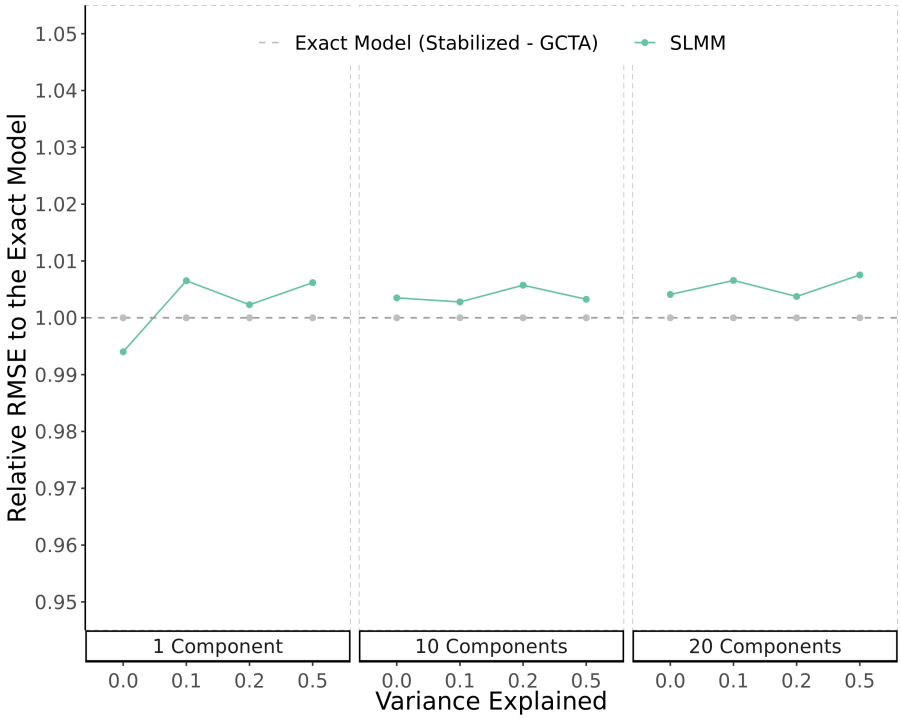
Supplementary Figure 5. Runtime of out-of-core (memory-friendly) training of REML using SLMM. Runtimes are for fitting a model with two components (even and odd autosomes) with varying sample size from the UK Biobank data (white British) with approx. 415K variants. (a) Single machine runtime on nodes with varying amounts of total memory (green and orange) along with a runtime of in-memory training (model uses approx. 30GB memory on 300K samples). Max 4 means only 4 cores used. 8GB or 64GB is the total system memory. (b) Memory footprint of the out-of-core runs through the model training. The high level of fluctuations in the run with less total memory (green line) is due to less cache usage by the OS which is likely the reason for slower runtime.



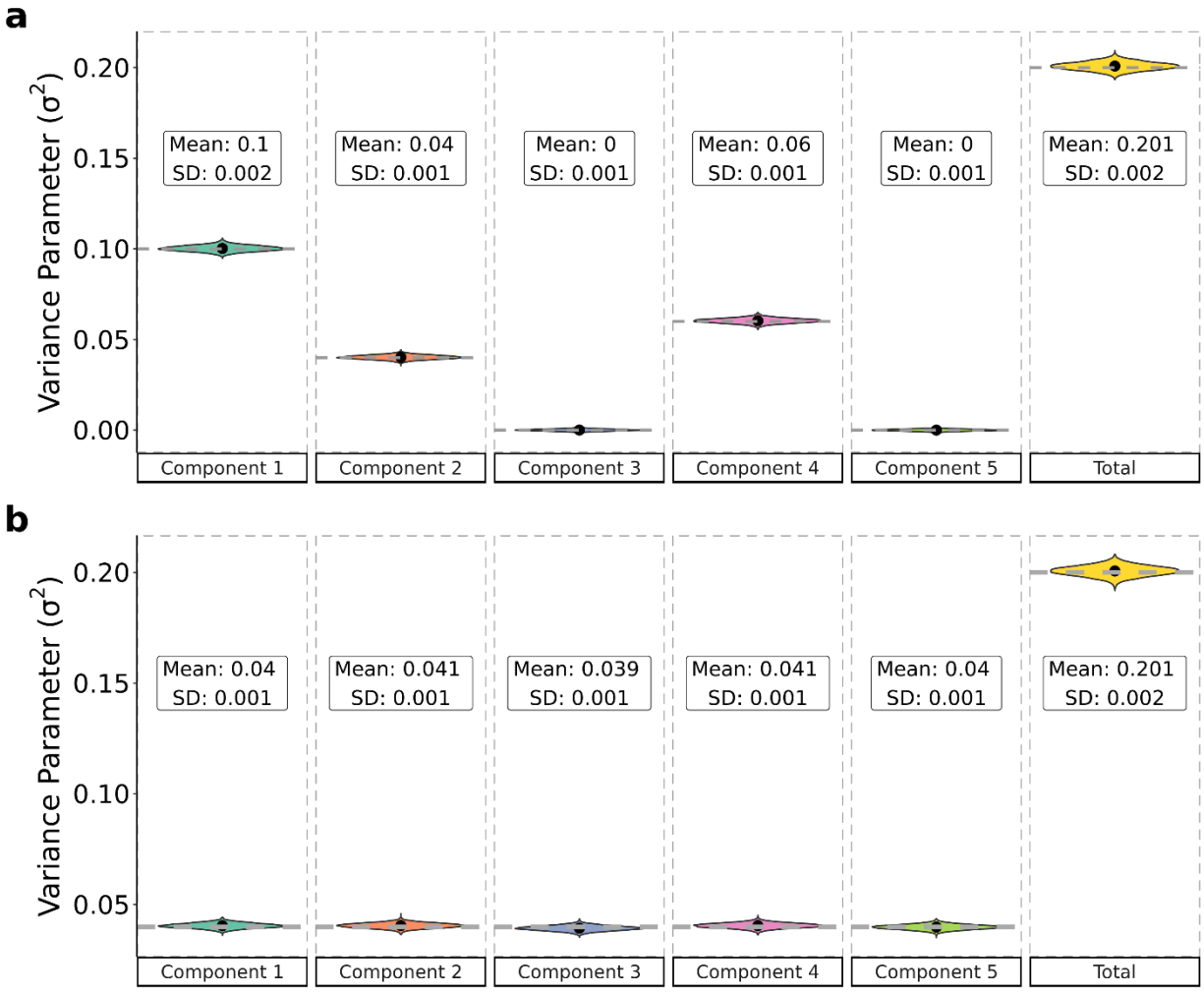
Supplementary Figure 6. Simulation results on 10K subset (10,000 subset of the UK Biobank white British samples with approx. 415K variants) with varying total variance explained (i.e., total SNP heritability across all components). The number of components is set to 20 representing the first 20 chromosomes. Points represent mean and bars represent one standard deviation from the mean. Numbers in boxes above bars indicate bar height. The number of simulations in each setting is 1,000. Note the bias of BOLT-LMM. To our understanding, this bias near zero is caused by constrained optimization (forcing positivity of estimates). This constraint-induced bias could potentially shrink as the sample size grows i.e., 100s of thousands of samples, however, BOLT-LMM's runtime performance scales poorly with such increase in sample size. Also note that SLMM can easily handle constraints for parameter estimates, meaning that one can set a lower bound for estimates, if needed, by adjusting the lower bound parameter in the fitting function call.



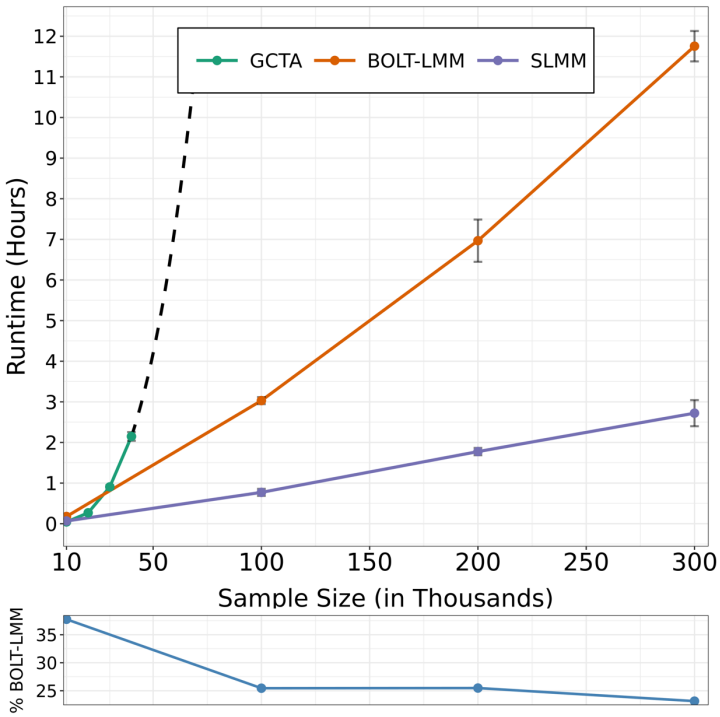
Supplementary Figure 7. SLMM is accurate compared to the exact model, as shown by simulation results representing per component estimates on a 10K subset (10,000 subset of the UK Biobank white British samples with approx. 415K variants). The total number of simulations is 1000. Points represent mean and bars represent one standard deviation from the mean.



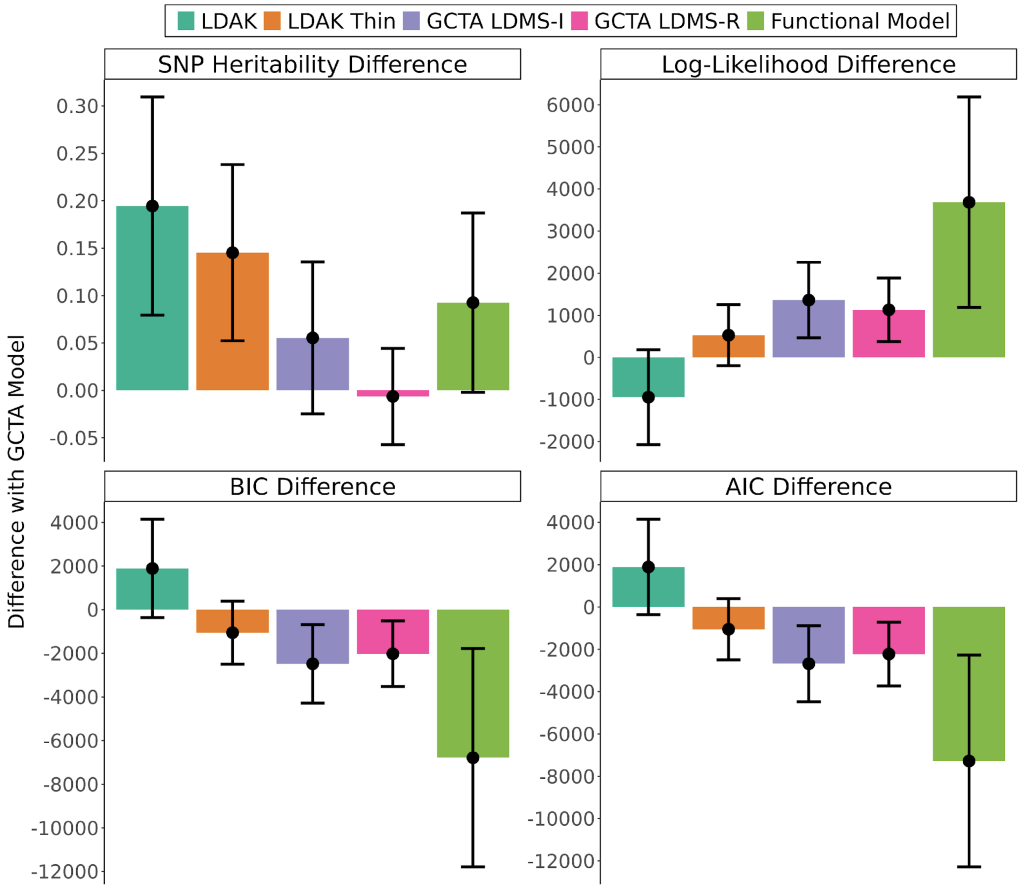
Supplementary Figure 8. SLMM shows little loss of statistical efficiency compared to the GCTA method. Presented is the relative Root Mean Squared Error of SLMM compared to that of GCTA. RMSE is calculated from the extensive simulation results presented above by varying both the variance explained and the number of components. In an ideal scenario, SLMM's RMSE would match that of GCTA (therefore the relative RMSE would be one), however, due to the use of randomized trace estimation (even with variance reduction), there is some loss of statistical efficiency in SLMM (increase in variance of the estimate, albeit small). Such a small increase in the variance of the estimate (approx. 1% increase in variance) could be thought of as the cost for significant decrease in the asymptotic complexity (faster runtime) of SLMM versus GCTA.



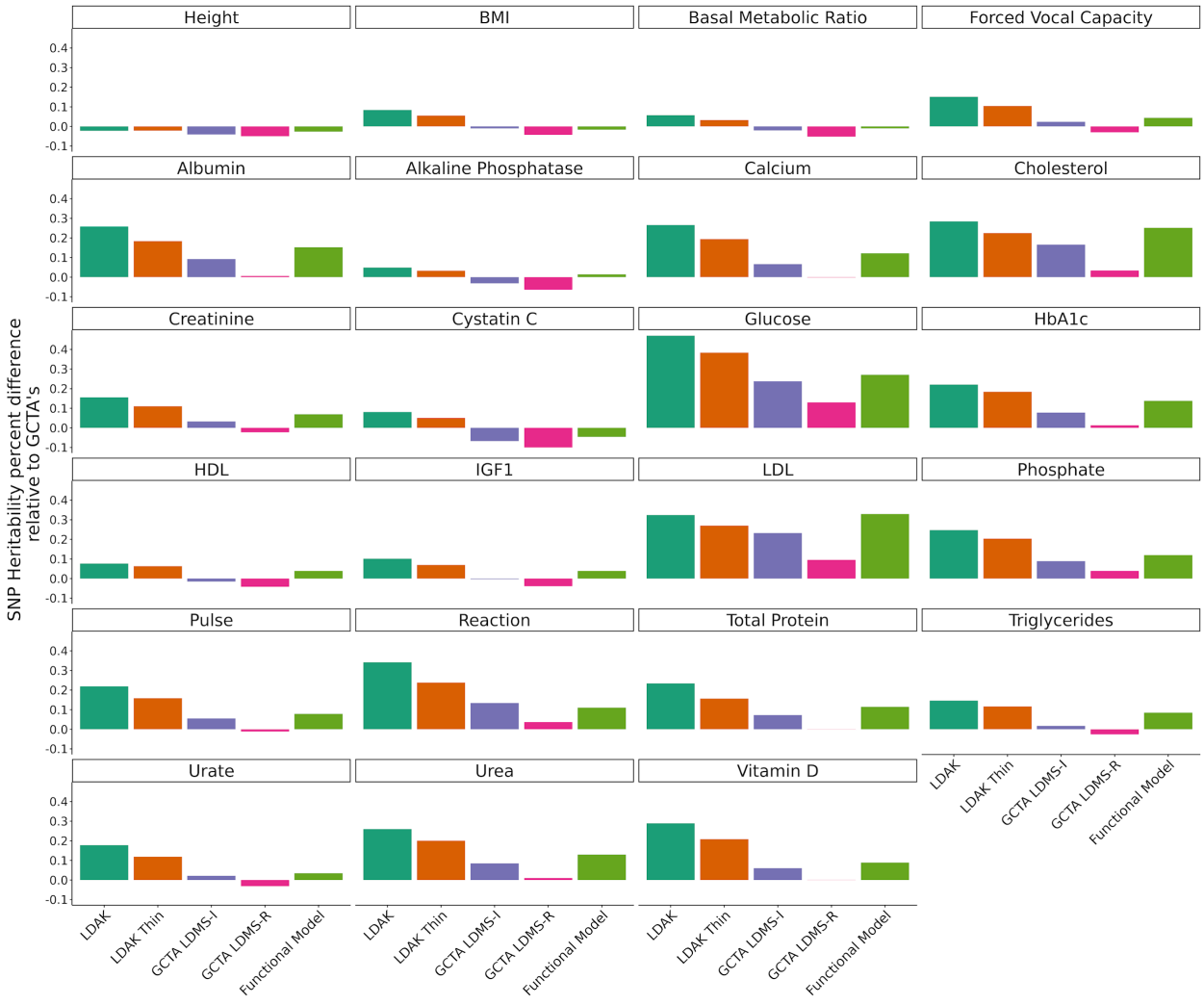
Supplementary Figure 9 Biobank-scale (300K samples, 146K variants on first 5 chromosomes) simulations illustrate the statistical accuracy of SLMM. The number of simulations is 100. Points represent the mean of estimate across all rounds of simulation. There are 5 components in total, one per chromosome. In the top plot, the total variance explained is set to 20% varying the components' share of variance explained. Bottom plot represents a scenario with no variance explained by any of the components. Some differences between the mean and true value are expected due to the number of simulations and would be expected to be reduced with more simulations.



Supplementary Figure 10. Runtime comparison of SLMM versus BOLT-LMM (REML) and GCTA. Top plot represents runtime for a model with 22 components (one per autosome) on a single machine with 40 CPUs. Bottom plot shows the relative decrease in runtime compared to that of BOLT-LMM (REML). BOLT does not compute likelihood and cannot handle non-GCTA models. It also does not scale to multiple machines for distributed computation.



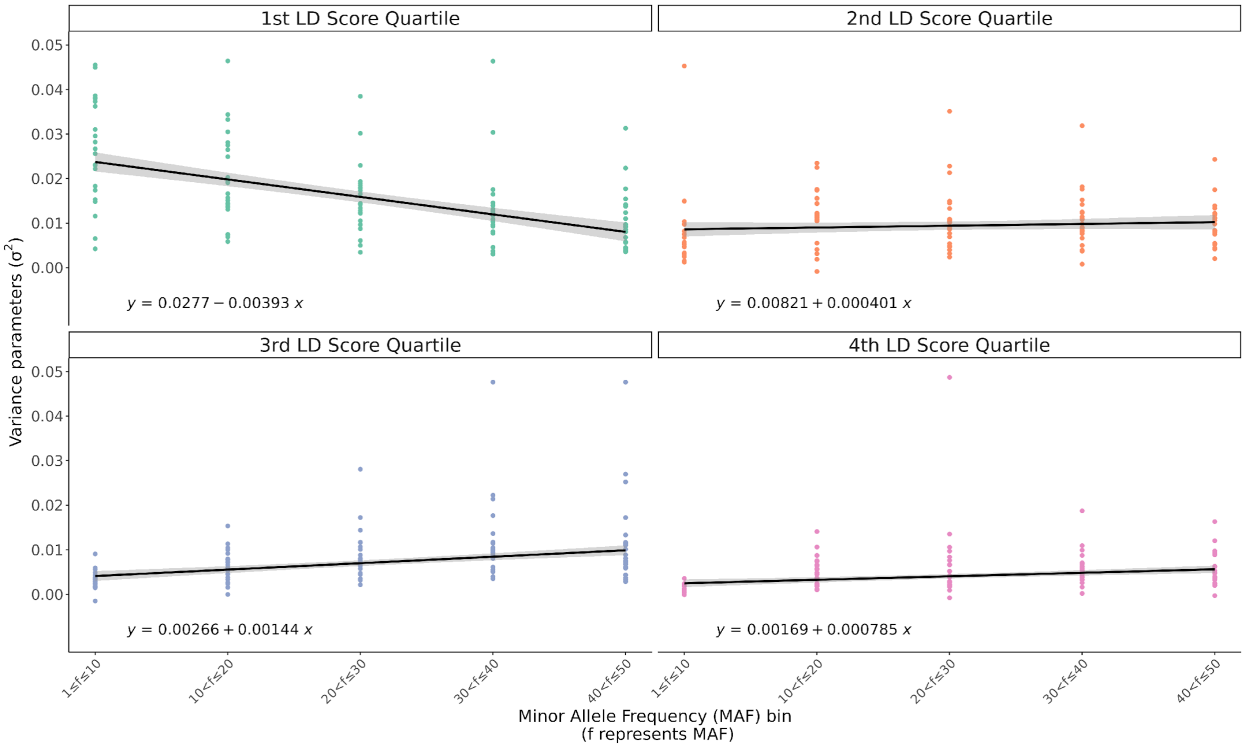
Supplementary Figure 11. Summary of heritability model comparison. Panels represent average (±standard deviation) difference between alternative heritability model and that of GCTA GREML on SNP heritability, log marginal likelihood, BIC and AIC.



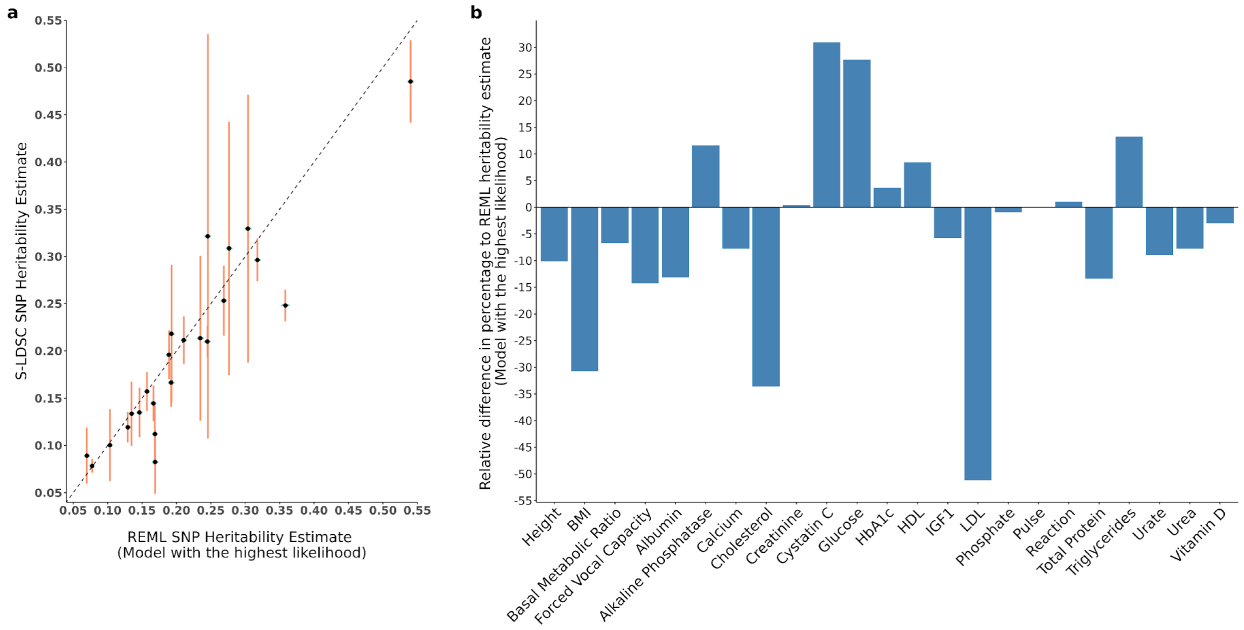
Supplementary Figure 12. Percent change in the SNP heritability estimates of the alternative heritability models across 23 phenotypes compared to the GCTA model. (Observed difference of alternative model's estimate minus GCTA is presented as a percent of GCTA's estimate). Panels represent differences per phenotype. Models include LDAK, LDAK-Thin, GCTA LDMS-I and GCTA LDMS-R and the Functional model which is the extension of the best model (GCTA LDMS-I) to include 27 additional components, one per SNP-to-annotation mapping.

Supplementary Figure 13. Change in the likelihood (log marginal likelihood of fitted LMM) across the 23 phenotypes compared to the GCTA model. (Alternative models' log marginal likelihood minus GCTA's, larger is better). Panels represent differences per phenotype. Models include LDAK, LDAK-Thin, GCTA LDMS-I and GCTA LDMS-R and the Functional model which is the extension of the best model (GCTA LDMS-I) to include 27 additional components, one per SNP-to-annotation mapping.

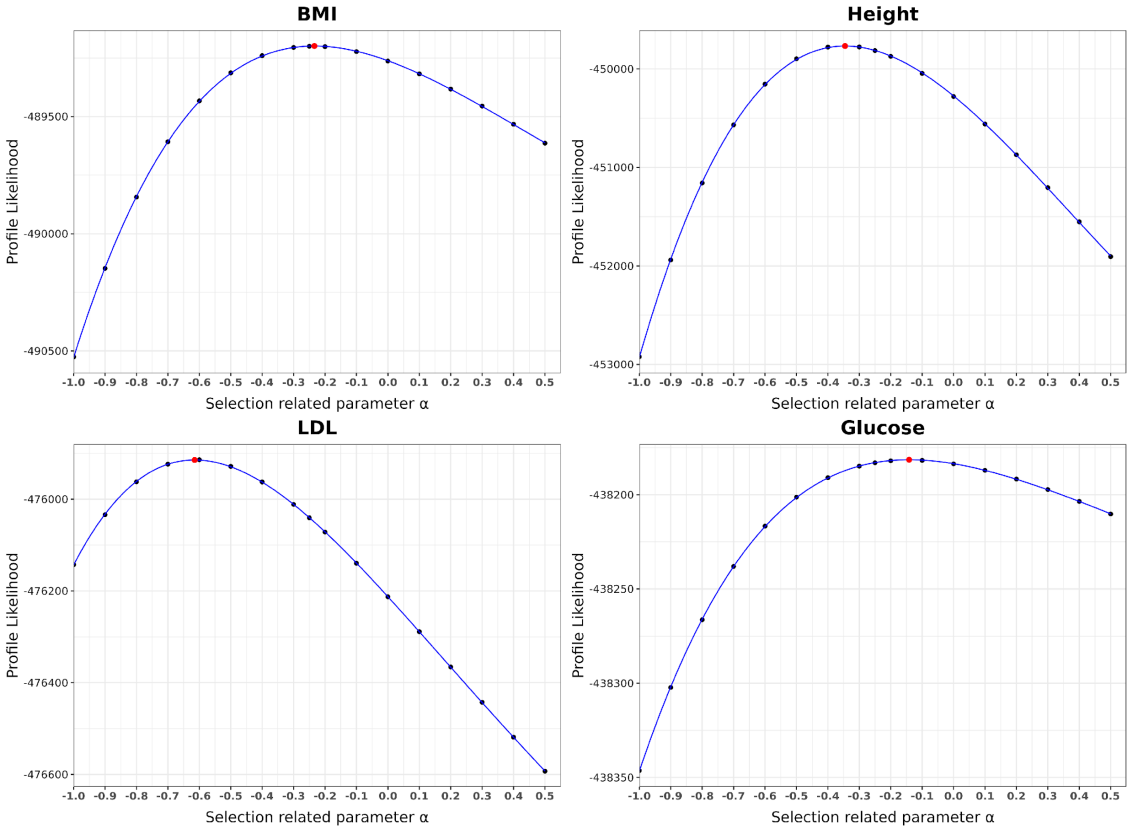
Supplementary Figure 14. Change in BIC (Bayesian Information Criterion) across the 23 phenotypes compared to the GCTA model. Alternative models' BIC minus GCTA's, smaller is better. Panels represent differences per phenotype. Models include LDAK, LDAK-Thin, GCTA LDMS-I and GCTA LDMS-R and the Functional model which is the extension of the best model (GCTA LDMS-I) to include 27 additional components, one per SNP-to-annotation mapping.



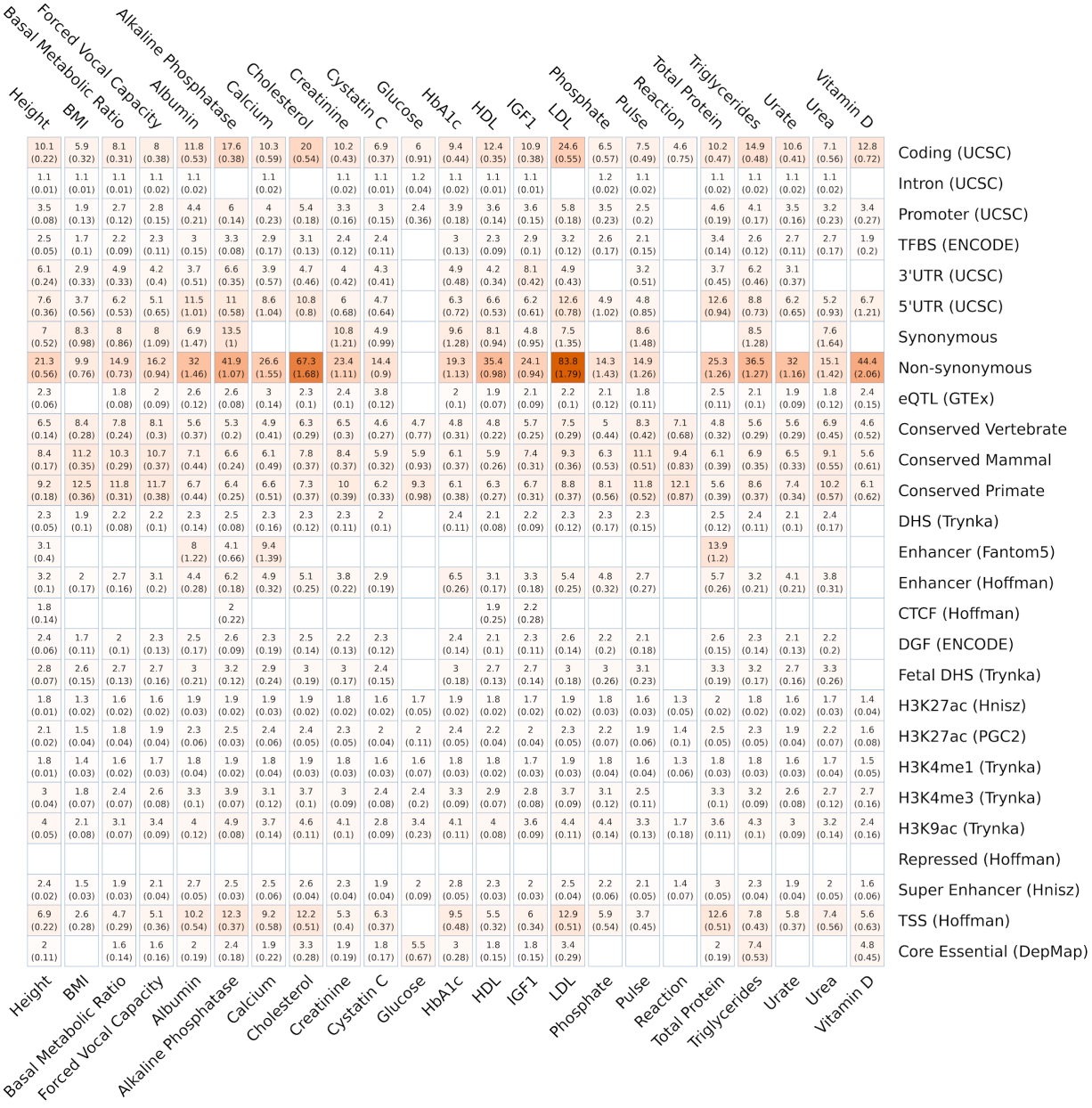
Supplementary Figure 15. LD and MAF dependent variation in variance estimates. Presented are the variance parameters of the GCTA LDMS-I heritability model (20 parameters) across all 23 phenotypes. An LDMS-I model contains 20 components that split the genome into 4 LD quartiles (based on SNP LD scores, first quartile is the lowest) and further split each quartile to 5 groups based on MAF. In this figure, each panel represents an LD quartile and therefore there are five points (representing MAF bins within LD quartile) per phenotype. There are 23 phenotypes in total. Fitted lines are from robust regression, and an estimated model of the trend line is presented in each panel (y represents variance estimate in each category and x represents the MAF bins encoded as an ordinal value).



Supplementary Figure 16. Comparison of heritability estimates from the best REML model to those from the baseline LD model (Stratified LDSC, S-LDSC). (a) Estimates ($\pm 2 \times$ standard errors) from REML versus S-LDSC model. Vertical orange lines represent standard error of the S-LDSC model and horizontal green lines (too small to be noticeable) represent standard error of the REML model. (b) The relative percentage difference in estimates between S-LDSC and the best REML model for a range of phenotypes. The estimates from S-LDSC are from a superset of variants than those used in this study (Info Score ≥ 0.9 and MAF ≥ 0.001 while we used Info Score ≥ 0.99 and MAF ≥ 0.01). We did not repeat the process on our subset of the data as we found their estimates from the larger set of variants are often smaller than ours and their estimates have extremely large standard errors.



Supplementary Figure 17. Example profile likelihood plots for estimating the selection related parameter α. Black dots represent grid points over which the LDAK-Thin model is fitted for each phenotype. Blue lines represent the polynomial approximation of the profile likelihood surface. Red points represent the interpolated maximum of the given profile likelihood. The profile likelihood confidence intervals are calculated based on the red points and the interpolated surface.



Supplementary Figure 18. Heatmap of functional heritability enrichment across 23 phenotypes. The enrichments are estimated through a joint model by adding all the 27 functional categories to the best model (GCTA LDMS-I). The functional categories are derived from the Baseline LD model v2.2 with the addition of an extra category representing Core Essential genes (see Methods). Numbers in the cells represent statistically significant (Bonferroni corrected: $\boldsymbol{P \leq 0.05/(27 \times 23)}$ tests) enrichments.