**LDAK-KVIK performs fast and powerful mixed-model association analysis of quantitative and binary phenotypes — Supplementary Material**

Jasper Hof[1] and Doug Speed[2*]
1 - Radboud University Medical Center, IQ Health Science Department, Nijmegen, The Netherlands
2 - Aarhus University, Center for Quantitative Genetics and Genomics, Aarhus, Denmark
* Corresponding author, doug@qgg.au.dk

# Supplementary Notes

# Supplementary Figures



# Supplementary Tables

# Supplementary Note 1: Example LDAK-KVIK commands

Here we provide a simple example of how to run LDAK-KVIK; for more extensive documentation, see the LDAK website (`www.dougspeed.com`). The following commands assume you have downloaded the Linux version of LDAK 6.1 (but note that we also provide a MAC version). Further, they assume that all data files are stored in PLINK format: specifically, that SNP genotypes are stored in the files `data.bed`, `data.bim` and `data.fam`, that measurements for a quantitative phenotype are stored in the file `data.pheno`, and that covariates are stored in the file `data.covar`. Finally, if performing a gene-based analysis (Step 3), you require gene annotations; this example uses the file `RefSeq_GRCh37.txt`, which can be downloaded from `www.dougspeed.com/resources`.

```
#Step 1 - Construct the LOCO PRS and estimate the test statistic scaling factor
./ldak6.1.linux --kvik-step1 kvik --bfile data --pheno data.pheno --covar data.covar

#Step 2 - Perform single-SNP association analysis
./ldak6.1.linux --kvik-step2 kvik --bfile data --pheno data.pheno --covar data.covar

#Step 3 - Perform gene-based association analysis
./ldak6.1.linux --kvik-step3 kvik --bfile data --genefile RefSeq_GRCh37.txt
```

Note that when analysing a binary phenotype, you should add `--binary YES` to Step 1.

The full results from single-SNP association analysis will be saved in the file `kvik.step2.assoc` (with a more concise version in `kvik.step2.summaries`). The results from the gene-based association analysis will be saved in the file `kvik.step3.remls.all`.

# Supplementary Note 2: Technical details of LDAK-KVIK

First we provide some definitions and summarize some existing methodology, then we explain the key innovations of LDAK-KVIK, and lastly we describe in detail the LDAK-KVIK algorithm. Note that most of this section assumes that the phenotype is quantitative, then at the end we explain the modifications used when the phenotype is binary.

## 2.1 Definitions and existing methodology

We assume there are $n$ individuals, each genotyped for $m$ SNPs, recorded for $q$ covariates and measured for a quantitative phenotype. Let the $(n \times m)$ matrix $X'$ contain the genotypes, let the length-$n$ vector $Y'$ contain phenotypes, and let the $(n \times q)$ matrix $Z$ contain covariates. Note that we always include an intercept, so that $q \geq 1$ and $Z_{i,1} = 1$ $\forall i$. We use $C$ to denote the total number of chromosomes, use $X$ and $Y$ to denote, respectively, the genotypes and phenotypes after regressing out the covariates (via ordinary least-squares regression), and use $\lambda$ to denote the test statistic scaling factor. Without loss of generality, we standardize $X_j$, the $j$th column of $X$, to have mean zero and variance one. We similarly standardize $Y$.

Note that the regression models in this section describe how the residual phenotypes $Y$ depend on the residual genotypes $X$. In general, these regressions are equivalent to instead modelling how the original phenotypes $Y'$ depend on the original genotypes $X'$ and including covariates $Z$ (see Supplementary Note 3 for a justification).

**Linear random-effects regression model.** LDAK-KVIK repeatedly uses models of the form

$$Y = X_1\gamma_1 + ... + X_m\gamma_m + e = X\gamma + e, \tag{1}$$

where $\gamma$ and $e$ are, respectively, length-$m$ and length-$n$ vectors of random effects representing SNP effect sizes and environmental noise contributions. The models vary according to the assumed prior distributions for $\gamma$ and $e$. For example, Ridge Regression Models use prior distributions of the form

$$\gamma_j \sim N(0, a_j), \quad \text{and} \quad e_i \sim N(0, \sigma_e^2),$$

whereas Elastic Net Models use prior distributions of the form

$$\gamma_j \sim pDE(b_j) + (1-p)N(0, c_j), \quad \text{and} \quad e_i \sim N(0, \sigma_e^2),$$

where $DE(b_j)$ denotes a double exponential distribution with rate parameter $b_j$. We solve the linear random-effects regression models (i.e., estimate $\gamma$) using variational Bayes. We provide a brief description of variational Bayes in the Online Methods, while a detailed description is provided in the Supplementary Material of the BOLT-LMM publication [1].

**Heritability models.** The heritability model describes how $h_j^2$, the proportion of phenotypic variance explained by SNP $j$ is expected to vary across the genome [2]. If $f_j$ denotes the minor allele frequency of SNP $j$, then LDAK-KVIK uses heritability models of the form

$$\mathbb{E}[h_j^2] = w_j h^2/W, \quad \text{where} \quad w_j = [f_j(1-f_j)]^{1+\alpha} \quad \text{and} \quad W = \sum_{j=1}^{m} w_j. \tag{2}$$

The parameter $h^2$ is the phenotypic variance explained by all SNPs ($h^2 = \sum \mathbb{E}[h_j^2]$); when the data are homogeneous, $h^2$ equals the SNP heritability, but for heterogeneous data, $h^2$ is intermediate of the SNP heritability and family heritability. [3–5]. The parameter $\alpha$ determines the relationship between the expected per-SNP heritabilities and MAF; if $\alpha > -1$, then $\mathbb{E}[h_j^2]$ increases with $f_j$, indicating that more common SNPs tend to contribute more heritabilities than less common SNPs, and vice versa [4]. Setting $\alpha = -1$ corresponds to assuming that $\mathbb{E}[h_j^2]$ is constant; we refer to this as the Uniform Heritability Model, and it is the model (implicitly) assumed by most software in statistical genetics (including BOLT-LMM [1], REGENIE [6], fastGWA [7] and GCTA-LOCO [8]). By contrast, our previous works, and those of others, have shown that values of $\alpha$ close to -0.25 are usually more appropriate when analyzing complex human traits [9–13].

Note that when using the linear random-effects regression model defined in Equation 1, the expected per-SNP heritability is $\mathbb{E}[h_j^2] = \mathbb{E}[\gamma_j^2] \times Var(X_j)/Var(Y) = \mathbb{E}[\gamma_j^2]$. Therefore, we can incorporate different heritability models by varying the prior distribution parameters. For example, if using a Ridge Regression Model, we can incorporate the heritability model defined in Equation 2 by setting $a_j = w_j h^2/W$ and $\sigma_e^2 = 1 - h^2$ (because then $\mathbb{E}[h_j^2] = \mathbb{E}[\gamma_j^2] = w_j h^2/W$, as desired).

When using a linear random-effects regression model, the choice of heritability model determines the expected covariance between elements of $Y$. For example, if we assume the heritability model defined in Equation 2, and let $\Omega$ denote an $m \times m$ diagonal matrix such that $\Omega_{j,j} = w_j / \sum w_j$, then the assumed phenotypic variance matrix is $V = Kh^2 + I(1-h^2)$, where the genomic relatedness matrix $K = X\Omega X^T$.

**Partitioned randomized Haseman-Elston Regression.** Haseman-Elston Regression is a method-of-moment approach for estimating heritability, that searches for parameter values so that the expected phenotypic variance matrix most closely matches the observed phenotypic variance matrix [14]. First we describe the basic version of Haseman-Elston Regression, then we explain how it can be randomized and partitioned.

As explained above, the expected phenotypic variance matrix is determined by the heritability model, and takes the form $V = Kh^2 + I(1-h^2)$. Meanwhile, the observed variance matrix is $O = YY^T$ (note that $Y$ has mean zero, so does not need to be centred). We can estimate $h^2$ by regressing the upper-triangular values of values of $O$ onto the corresponding upper-triangular values of $K$ (note that it suffices to ignore the diagonal values because both $Tr(O)$ and $Tr(K)$, the traces of $O$ and $K$, respectively, are very close to $n$, while it suffices to ignore the lower-triangular values because both $O$ and $K$ are symmetric). The ordinary least-squares estimate of $h^2$ is

$$\hat{h}^2 = (K_U^T K_U)^{-1}(K_U^T O_U),$$

where $K_U$ and $O_U$ are vectors containing the upper-triangular values of $K$ and $O$, respectively.

Instead of computing $K_U^T K_U$ directly, randomized Haseman-Elston Regression writes this term as $K_U^T K_U = (Tr(KK) - \sum_i K_{i,i}^2)/2$, then uses the Monte Carlo approximation $Tr(KK) = \sum_r v_r^T KKv_r/R$, where $v_1, \ldots, v_R$ are length-$n$ random vectors with expected mean zero and expected variance one (in practice, the elements of each $v_r$ are sampled from a standard normal distribution). Note that with randomized Haseman-Elston Regression, it is not necessary to store all genotypes in memory. This is because if we divide the genome into $D$ chunks, we can write $Kv_r = X\Omega X^T v_r = X_1\Omega_1 X_1^T v_r + \ldots + X_D\Omega_D X_D^T v_r$, where $X_d$ contains the genotypes for the $d$th chunk, and $\Omega_d$ is the corresponding sub-matrix of $\Omega$. Therefore, it suffices to read the data one chunk at a time, which substantially reduces memory requirements.

The partitioned version of Haseman-Elston Regression arises from generalizing the heritability model. For example, suppose we divide the genome into $B$ partitions, then assume

$$\mathbb{E}[h_j^2] = I(j,1)w_j H_1/W_1 + \ldots + I(j,B)w_j H_B/W_B, \quad \text{where} \quad W_b = \sum_j I(j,b)w_j. \tag{3}$$

Here, $I(j,b)$ is an indicator function that equals one (zero) if SNP $j$ is inside (outside) Partition $b$, while $H_b$ is the heritability of Partition $b$. The expected phenotypic variance matrix now takes the form $V = K_1 H_1 + \ldots + K_B H_B + I(1-h^2)$, where $K_b$ is a genomic relatedness matrix computed using only SNPs in Partition $b$. Partitioned Haseman-Elston Regression estimates $H_1, H_2, \ldots, H_B$ (and so also their sum, $h^2$), by regressing the upper-triangular values of $O$ jointly on the corresponding upper-triangular values of $K_1, K_2, \ldots, K_B$.

Note that partitioned Haseman-Elston Regression has previously been advocated because it enables more realistic heritability models [14]. However, our motivation for using it is slightly different. Namely, we observe that

the accuracy of basic Haseman-Elston Regression depends on the magnitude of $h^2$ (it tends to be more accurate when $h^2$ is small, and less accurate when $h^2$ is large). To understand why, note that the estimate of $h^2$ from basic Haseman-Elston Regression is equivalent to that you would obtain from one-step zero-start Newton-Raphson REML (i.e., from running Newton-Raphson REML if you required that the starting estimate of $h^2$ is zero, and stop after only one iteration). When $h^2$ is close to zero, one-step zero-start Newton-Raphson REML will likely be reasonably accurate (because the algorithm does not need to move far). However, when $h^2$ is moderate or large, one-step zero-start Newton-Raphson REML can perform poorly (because it needs to make a very large jump). It follows that the accuracy of partitioned Haseman-Elston Regression will depend on the magnitude of each $H_b$. However, provided we use sufficient partitions, $H_b$ will tend to be small, and therefore partitioned Haseman-Elston Regression will tend to be accurate.

**Monte Carlo REML.** Consider the mixed model $Y \sim N(0, Kh^2 + I(1-h^2))$. The REML estimate of $h^2$ maximizes

$$l(Y|h^2) = -\frac{1}{2}Y^T V^{-1} Y - \frac{1}{2}\log|V|, \quad \text{where} \quad V = Kh^2 + I(1-h^2).$$

We can find the maximum by treating the likelihood as a function of $h^2$ and $1-h^2$ (even though the two parameters are linearly dependent), then setting the (partial) derivatives with respect to $h^2$ and $1-h^2$ to zero:

$$\frac{dl}{dh^2} = 0 \Rightarrow Y^T V^{-1} K V^{-1} Y = Tr(KV^{-1}) \quad \text{while} \quad \frac{dl}{d(1-h^2)} = 0 \Rightarrow Y^T V^{-1} V^{-1} Y = Tr(V^{-1}).$$

It follows that $\hat{h}^2$, the REML estimate of $h^2$, satisfies

$$\delta(\hat{h}^2) = \frac{Y^T V^{-1} K V^{-1} Y}{Y^T V^{-1} V^{-1} Y} \Big/ \frac{Tr(KV^{-1})}{Tr(V^{-1})} = 1.$$

As noted by the authors of BOLT-LMM, [1] the task of finding $\hat{h}^2$ is simplified by the fact that $\delta(h^2)$ tends to vary monotonically with $h^2$. Instead of computing the traces exactly, we can use the Monte Carlo estimates $Tr(KV^{-1}) = \sum_r v_r^T K V^{-1} v_r / R$ and $Tr(V^{-1}) = \sum_r v_r^T V^{-1} v_r / R$, where $v_1, \ldots, v_R$ are length-$n$ random vectors with expected mean zero and expected variance one (the same strategy utilized by randomized Haseman-Elston Regression).

**The two-step MMAA approach.** In the following paragraphs, we explain the rationale behind the two-step MMAA approach (e.g., that used by BOLT-LMM and REGENIE [1,6]). Note that, for mathematical simplicity, we ignore the fact that most modern MMAA tools use LOCO (e.g., we treat the estimated variance matrix $\hat{V}$ and the PRS $P$ as fixed, when in practice, these will usually vary depending on the chromosome being tested).

Early MMAA tools (e.g., EMMA, GEMMA and FastLMM [15–17]) tested SNPs using models of the form

$$Y \sim N(X_j \beta_j, \hat{V}\sigma^2) \quad \text{with} \quad \hat{V} = K\hat{h}^2 + I(1-\hat{h}^2),$$

where $X_j$ is the SNP being tested and $\beta_j$ is its effect size. The generalized least-squares estimate of $\beta_j$ is

$$\hat{\beta}_j = \frac{X_j^T \hat{V}^{-1} Y}{X_j^T \hat{V}^{-1} X_j},$$

whose estimated variance is

$$Var(\hat{\beta}_j) = \frac{\hat{\sigma}^2}{(X_j^T \hat{V}^{-1} X_j)} \quad \text{where} \quad \hat{\sigma}^2 = \frac{Y^T \hat{V}^{-1} Y}{n-q},$$

and thus a $\chi^2(1)$ test statistic is

$$S_j = \frac{\hat{\beta}_j^2}{Var(\hat{\beta}_j)} = \frac{(X_j^T \hat{V}^{-1} Y)^2}{X_j^T \hat{V}^{-1} X_j \times \hat{\sigma}^2}.$$

Note that many existing two-step MMAA tools exclude $\sigma^2$, reflecting that if $\hat{V}$ accurately models the variance of $Y$, then $\sigma^2 = 1$ (and so its estimate can be ignored). However, we prefer to include this term, because we believe it guards against misspecification of $\hat{V}$ (e.g., imperfect estimation of $h^2$). Note also that the inclusion of $q$ in the denominator of $\hat{\sigma}^2$ reflects that we are using residual genotypes and phenotypes.

The early MMAA tools were computationally demanding due to the fact that when calculating the generalized least-squares estimates, they would explicitly compute $\hat{V}$, then its inverse. However, later software (e.g., GRAMMAR [18] and BOLT-LMM [1]), recognised that an equivalent (but much more efficient) alternative is to compute a PRS assuming the Ridge Regression Model, then perform ordinary least-squares regression using the PRS as an offset. Although the resulting test statistics may be biased, this can be corrected for by scaling the test statistics by a constant.

Specifically, the two-step approach works as follows. Suppose we assume the Ridge Regression Model $Y = X\gamma + e$, with $\gamma_j \sim N(0, \hat{h}_j^2)$ and $e_i \sim N(0, 1 - \hat{h}^2)$, then construct the PRS $P = X\hat{\gamma}$, where $\hat{\gamma}$ is the posterior mean of $\gamma$. It can be shown that $\hat{\gamma} = \hat{h}^2 K \hat{V}^{-1} Y$, and in turn that $Y - P = (1 - \hat{h}^2)\hat{V}^{-1} Y$ (see Supplementary Note 4 for a proof). An (unscaled) $\chi^2(1)$ test statistic from regressing $Y - P$ on $X_j$ is

$$T_j = \frac{(X_j^T \hat{V}^{-1} Y)^2}{X_j^T X_j \times \hat{s}^2} \quad \text{with} \quad \hat{s}^2 = \frac{Y^T \hat{V}^{-1} \hat{V}^{-1} Y}{n - q},$$

where $\hat{s}^2$ arises as an unbiased estimate of $Var(\hat{V}^{-1} Y)$. We can therefore write

$$S_j = \lambda_j' T_j, \quad \text{where} \quad \lambda_j' = \frac{X_j^T X_j \times \hat{s}^2}{X_j^T \hat{V}^{-1} X_j \times \hat{\sigma}^2}.$$

Empirical studies have found that $\lambda_j'$ tends to be almost constant across the genome [1, 18], and can therefore be replaced by $\lambda'$, the mean value of $\lambda_j'$ computed across a small number of randomly-picked SNPs (we refer to this as the "Grammar-Gamma Formula").

The authors of BOLT-LMM realised that the two-step approach described above can be improved by using a mixture prior for SNP effect sizes. They therefore proposed assuming the linear random-effects regression model

$$Y = X\gamma + e, \quad \text{with} \quad \gamma_j \sim pN\left(0, \frac{1-F}{p} \times \frac{\hat{h}^2}{m}\right) + (1-p)N\left(0, \frac{F}{1-p} \times \frac{\hat{h}^2}{m}\right) \quad \text{and} \quad e_i \sim N(0, 1 - \hat{h}^2),$$

where the hyperparameters $p$ and $F$ are set via cross-validation. Let $U_j$ denote the resulting test statistics (i.e., those obtained by computing $P = X\hat{\gamma}$, where $\hat{\gamma}$ is the revised posterior mean of $\gamma$, then regressing $Y - P$ on each $X_j$). Like $T_j$, these test statistics must be scaled by a constant, which we denote $\lambda$.

BOLT-LMM estimates $\lambda$ using a tool called LD Score Regression (LDSC). [19] BOLT-LMM first runs LDSC using the scaled test statistics $\lambda' T_1, \ldots, \lambda' T_m$, where $T_1, \ldots, T_m$ are obtained by performing the two-step MMAA approach using a Ridge Regression PRS, and $\lambda'$ is estimated via the Grammar-Gamma Formula. BOLT-LMM then runs LDSC using the unscaled test statistics $U_1, \ldots, U_m$. If $A_1$ and $A_2$ denote the estimated intercepts from the two

analyses, then BOLT-LMM set $\lambda = A_1/A_2$ (i.e., this value ensures that the two sets of scaled test statistics have the same estimated intercept).

**fastGWA.** When applied to quantitative phenotypes, fastGWA [7] performs generalized least-squares regression using the model $Y' \sim N(Z\theta + X'_j\beta_j, \hat{V}_S)$, where $\hat{V}_S$ is a sparse approximation of the estimated genome variance matrix $\hat{V}$ (the authors of fastGWA recommend obtaining $\hat{V}_S$ by setting elements of $\hat{V}$ below 0.05 to zero). [7] The sparsity of $\hat{V}_S$ enables fastGWA to rapidly compute $\hat{V}_S^{-1}Y$ and terms of the form $\hat{V}_S^{-1}X_j$, that are required when estimating SNP effect sizes and the corresponding variances. fastGWA reduces runtime further by using the GRAMMAR-Gamma Formula.

When applied to binary phenotypes, fastGWA switches from a linear mixed model to a logistic mixed model, then solves the latter by maximizing the quasi-likelihood. [20, 21] fastGWA continues to require that the variance matrix is sparse, and uses a modified version of the GRAMMAR-Gamma Formula.

Note that LDAK-KVIK includes an approximate version of fastGWA (described in the Online Methods), that is used when analyzing binary phenotypes for datasets determined to have strong structure.

**LDAK-GBAT.** We have previously developed LDAK-GBAT, a tool for gene-based association analysis using GWAS summary statistics and a reference panel [22]. LDAK-GBAT uses REML to solve the model

$$Y \sim N(0, K_S\sigma_S^2 + I(1-\sigma_S^2)),$$

where $K_S$ is a "genomic" relatedness matrix constructed using only SNPs within the gene being tested, then performs a likelihood ratio test of $\sigma_S^2 > 0$ (using permutations to estimate the null distribution of the likelihood ratio test statistic). When applied to 109 phenotypes from the UK Biobank [23, 24], Million Veterans Program [25] and Psychiatric Genomics Consortium [26], LDAK-GBAT found at least 19% more significant genes than the existing tools MAGMA [27], fastBat [27], SKAT-O, PCA and ACAT (the last three tools are contained within the sumFREGAT software [28]).

Note that LDAK-GBAT requires a gene annotations file, which should be in Browser Extensible Data format [29]; i.e., it should contains one row for each gene, and four columns, that report the gene name and chromosome, and its start and end basepairs. For example, if the file contained a single line, with entries "ABC 7 0 10", this would indicate there is one gene, called ABC, which spans the first ten basepairs of Chromosome 7. For our gene-based association analyses of UK Biobank data, we tested 17 322 genes defined based on RefSeq annotations (the corresponding annotations file is provided at `www.dougspeed.com/resources`).

## 2.2 Key Innovations

LDAK-KVIK has five key innovations (note that these innovations are described in the Online Methods):

**1 - A novel variational Bayes Solver.** We have developed a chunk-based variational Bayes solver. This is not only used to construct PRS, but also to compute terms of the form $V^{-1}A$, where $V$ is a variance matrix and $A$ is

a length-$n$ vector. Supplementary Figure 2 shows that when constructing PRS, our variational Bayes solver is 5-20 times faster than a standard (genome-wide) variational Bayes solver, and that when calculating $V^{-1}A$, our solver is 3-10 times faster than conjugate gradient descent. Furthermore, our variational Bayes solver has a trivial memory footprint (less than 1 Gb), because it never needs to read in more than 256 SNPs at a time.

**2 - A novel SPA Solver.** We have developed an SPA solver, which is used by default in Step 2 when analyzing binary phenotypes. As shown in Supplementary Table 3, our solver has similar speed to fastSPA, the SPA solver implemented within REGENIE [6,30]. However, we believe our solver has two advantages. Firstly, it is empirical, so can be applied to all types of phenotype (by contrast, fastSPA assumes a Bernoulli Distribution, so can only be used for binary phenotypes). Secondly, our solver does not require that the genotypes are sparse (whereas the efficiency of fastSPA relies on relatively few individuals having non-zero genotypes for each SNP). We expect that future work can exploit these two advantages, for example, by applying our SPA solver to non-binary phenotypes (e.g., count, ordinal and survival phenotypes) or by applying our SPA solver to non-sparse predictors (e.g., genotype probabilities or transcriptomic measurements).

**3 - Incorporates more realistic heritability models.** Most existing MMAA tools assume that expected per-SNP heritabilities are constant, which is equivalent to assuming the heritability model defined in Equation 2 with $\alpha = -1$. Supplementary Figure 16 shows that when analyzing the 40 quantitative UK Biobank phenotypes, the power advantage of LDAK-KVIK over BOLT-LMM [1] is mainly because the former constructs more accurate LOCO PRS, which in turn is because LDAK-KVIK estimates $\alpha$ from the data, whereas BOLT-LMM fixes its value to -1.

**4 - Uses an Elastic Net prior distribution for SNP effect sizes.** While the use of the elastic net is fairly common in statistical genetics software [31, 32], we are not aware of it being used in MMAA tools. Supplementary Figure 16 shows that, when analyzing the 40 quantitative UK Biobank phenotypes, using the elastic net prior distribution provides a small advantage over using a mixture of two normal distributions (the assumption of Bolt-LMM [1]), and a large advantage over using a single normal distribution (similar to the assumption of REGENIE [6]).

**5 - A novel test for structure.** We have developed a test for heterogeneity, which LDAK-KVIK uses to determine which algorithms to use, and how to calculate the test statistic scaling factor $\lambda$. When LDAK-KVIK finds weak structure, then it constructs the Step 1 PRS assuming the elastic net prior distribution described above. However, when LDAK-KVIK finds strong structure, it either constructs the PRS assuming an infinitesimal prior distribution (quantitative phenotypes) or switches to an approximate version of fastGWA (binary phenotypes).

## 2.3 LDAK-KVIK Algorithm

LDAK-KVIK has two steps when used for single-SNP association analysis, and three steps when used for both single-SNP and gene-based association analysis.

**Step 1: Construct LOCO PRS and estimate $\lambda$**

Operation 1a - Test for structure.

Operation 1b - Estimate $\alpha$ and $h_j^2$ using partitioned randomized Haseman-Elston Regression.

Operation 1c - Revise the estimates of $h_j^2$ using Monte Carlo REML.

Operation 1d - Determine suitable elastic net hyperparameters via cross-validation*

Operation 1e - Construct LOCO PRS and estimate $\lambda$

* Note that Operation 1d is not required if Operation 1a determines there is strong structure.

**Step 2: Single-SNP association analysis**

Operation 2a - Calculate uncalibrated test statistics

Operation 2b - Scale test statistics

**Step 3: Gene-based association analysis**

Operation 3a - Run LDAK-GBAT using the results from single-SNP association analysis

First we describe each operation in turn, then we explain some implementation details. As a reminder, all regressions use residual genotypes and phenotypes, because this removes the need to include the covariate matrix $Z$ (see Supplementary Note 3 for a justification). The residual phenotypes (genotypes) are obtained by computing $Y = HY'$ ($X = HX'$), where $H = I - Z(Z^TZ)^{-1}Z^T$, then scaling $Y$ (columns of $X$) to have variance one.

**Operation 1a - Test for structure.** This test is fully described in Online Methods. However, in brief, LDAK-KVIK picks 512 SNPs semi-randomly from across the genome, then computes $\bar{\rho}^2$, the average squared correlation between pairs of SNPs on different chromosomes. LDAK-KVIK determines there is strong structure if both $\bar{\rho}^2$ is significantly greater than $1/(n-1)$, its expectation when the data are homogeneous, and $n\bar{\rho}^2 > 0.1$ (the latter is considered an estimate of the maximum average inflation of test statistics due to structure).

**Operation 1b - Estimate $\alpha$ and $h_j^2$ using partitioned randomized Haseman-Elston Regression.** LDAK-KVIK first obtains $\hat{\alpha}$ and $\hat{h}^2$, estimates of $\alpha$ and $h^2$, respectively, by running randomized Haseman-Elston Regression using the partitioned heritability model defined in Equation 3. LDAK-KVIK considers five different values for $\alpha$ (-1, -0.75, -0.5, -0.25 & 0), then sets $\hat{\alpha}$ to the value that results in best-fitting $V$ (measured by the sum of squared differences between off-diagonal elements of $V$ and $O$), and sets $\hat{h}^2$ to the corresponding estimate of $h^2$. LDAK-KVIK then obtains $\hat{h}_j^2$, estimates of the per-SNP heritabilties, by setting $\hat{h}_j^2 = w_j\hat{h}^2/W$, where $w_j = [f_j(1-f_j)]^{1+\hat{\alpha}}$.

By default, LDAK-KVIK uses 40 equally-sized partitions and either ten random vectors (if $n < 40\,000$) or three random vectors (if $n > 40\,000$). Supplementary Figure 24 shows that this operation produces reasonable estimates of both $h^2$ and $\alpha$.

**Operation 1c - Revise the estimates of $h_j^2$ using Monte Carlo REML.** LDAK-KVIK first evaluates $\delta(0.5\times\hat{h}^2)$, $\delta(0.75\times\hat{h}^2)$ and $\delta(\hat{h}^2)$, where $\hat{h}^2$ is the estimate of $h^2$ from randomized Haseman-Elston Regression (e.g., if the estimate of $h^2$ from randomized Haseman-Elston Regression is 0.4, then LDAK-KVIK evaluates $\delta(0.2)$, $\delta(0.3)$ and $\delta(0.4)$). LDAK-KVIK then uses linear interpolation or extrapolation to find an approximate solution to $\delta(h^2) = 1$. Finally, LDAK-KVIK recomputes the estimates of $h_j^2$ using the revised estimate of $h^2$ (continuing to use the estimate

of $\alpha$ from randomized Haseman-Elston Regression).

Supplementary Figure 24 shows that this operation reliably estimates $h^2$.

**Operation 1d - Determine suitable elastic net hyperparameters via cross-validation.** As a reminder, this operation is not required if Operation 1a determines there is strong structure. When constructing Elastic Net PRS, LDAK-KVIK assumes

$$\gamma_j \sim pDE\left(\left[\frac{2p}{(1-F)\hat{h}_j^2}\right]^{0.5}\right) + (1-p)N\left(0, \frac{F\hat{h}_j^2}{1-p}\right) \quad \text{and} \quad e \sim N(0, 1-\hat{h}^2). \tag{4}$$

In the prior distribution for SNP effect sizes, the parameter $p$ determines the contribution of the lasso component, while the parameter $F$ determines the expected contribution to variance from the ridge regression component. Note that this prior distribution is constructed so that $\mathbb{E}[h_j^2] = \hat{h}_j^2$ (i.e., the expected per-SNP heritabilities match their estimates from Operation 1c).

LDAK-KVIK uses cross-validation to obtain suitable values for $p$ and $F$. By default, it uses 90% of individuals to construct ten genome-wide PRS, for $(p, F)$ equal to (0.5,0.5), (0.5,0.3), (0.5,0.1), (0.1,0.5), (0.1,0.3), (0.1,0.1), (0.01,0.5), (0.01,0.3), (0.01,0.1) and (0,1). It then measures the accuracy of these PRS on the remaining 10% of individuals, selecting the values for $p$ and $F$ that result in the PRS with smallest mean-squared error. Note that if $MSE$ denotes the smallest mean-squared error, then LDAK-KVIK uses $n/MSE$ as an estimate of the effective sample size for the final association analysis (recall that $Var(Y) = 1$, so $1/MSE$ estimates the reduction in phenotypic variance when using the LOCO Elastic Net PRS as offsets).

**Operation 1e - Construct LOCO PRS and estimate $\lambda$.** The type of PRS and value of $\lambda$ depend on the outcome of the test for structure in Operation 1a. Suppose the test found only weak structure. Let $I(j,c)$ denote whether SNP $j$ is on Chromosome $c$, and let $S_c = (\sum_j I(j,c)\hat{h}_j^2)/\hat{h}^2$. LDAK-KVIK constructs $P_c$, the $c$th LOCO PRS, assuming the Elastic Net prior distribution for SNP effect sizes defined in Equation (4), with $p$ and $F$ set to the values identified in Operation 1d, and with $\hat{h}_j^2$ replaced by $\hat{h}_j^2 I(j,c)/S_c$ (note that multiplying the estimated per-SNP heritabilities by $I(j,c)/S_c$ ensures that they will be zero for SNPs on Chromosome $c$, but continue to sum to $\hat{h}^2$). LDAK-KVIK sets $\lambda = 1$.

When the test in Operation 1a finds strong structure, LDAK-KVAK instead constructs $P_c$ assuming the Ridge Regression prior distribution $\gamma_j \sim N(0, \hat{h}_j^2 I(j,c)/S_c)$, then estimates $\lambda$ using the Grammar-Gamma Formula (using, by default, 20 randomly-picked SNPs).

**Operation 2a - Calculate uncalibrated test statistics**. If SNP $j$ is on Chromosome $c$, then LDAK-KVIK tests it for association using ordinary least-squares regression with the model $\mathbb{E}[Y - P_c] = X_j\beta_j$, where $P_c$ is the LOCO PRS constructed in Operation 1e. The estimated effect size is

$$\hat{\beta}_j = \frac{X_j^T(Y - P_c)}{X_j^T X_j},$$

with estimated variance

$$Var(\hat{\beta}_j) = \frac{\hat{s}^2}{X_j^T X_j} \quad \text{where} \quad \hat{s}^2 = \frac{(Y - P_c)^T (Y - P_c)}{n - q},$$

and the corresponding $\chi^2(1)$ test statistic is

$$U_j = \frac{(X_j^T (Y - P_c))^2}{X_j^T X_j \times \hat{s}^2}.$$

**Operation 2b - Scale test statistics**. LDAK-KVIK reports three values for each SNP: an effect size estimate $\epsilon_1$, an estimate of the variance of the effect size estimate $\epsilon_2$, and a $\chi^2(1)$ test statistic $\epsilon_3$. LDAK-KVIK sets $\epsilon_1 = \lambda\hat{\beta}_j$ and $\epsilon_3 = \lambda U_j$, where $\hat{\beta}_j$ and $U_j$ are, respectively, the estimated effect size and test statistic calculated in Operation 2a, while $\lambda$ is the estimated scaling factor from Operation 1e. LDAK-KVIK then sets $\epsilon_2 = \epsilon_1^2/\epsilon_3$, which ensures that the three reported values are consistent (i.e., that $\epsilon_3 = \epsilon_1^2/\epsilon_2$).

**Operation 3a - Run LDAK-GBAT using the results from single-SNP association analysis**. LDAK-GBAT requires four inputs, a file containing gene annotations, a value for $\alpha$, GWAS summary statistics, and a reference panel. The gene annotations must be provided by the user (if analyzing human data, the user can download annotations from `www.dougspeed.com/resources`). LDAK-KVIK uses the estimate of $\alpha$ from Operation 1b and the summary statistics from Operation 2b. Finally, LDAK-KVIK randomly picks 5000 of the $n$ individuals, and uses their genotypes as an (in-sample) reference panel.

**Reducing runtime.** LDAK-KVIK minimizes memory usage by reading the SNP data on-the-fly. However, this means that a key determinant of its runtime is the number of times it must read each SNP once (we refer to this as the "number of scans"). For large datasets, the time to perform a scan is non-trivial (e.g., when analyzing 400 k individuals and 600 k SNPs, one scan takes about 0.5 CPU hours). Therefore, we have implemented LDAK-KVIK in a way that aims to minimize the total number of scans. For example, in Operation 1b, we run all the Haseman-Elston Regressions simultaneously, while we perform Operations 1c & 1d together. Furthermore, when performing Operations 1c & 1d, instead of waiting for convergence of the approximate log likelihood (the requirement in Operation 1e), we stop the Variational Bayes solver when the revised estimate of $h^2$ and the accuracy of the best-fitting PRS have both converged (changed by less than 0.005).

**Customizing LDAK-KVIK.** The above description of LDAK-KVIK assumed the default settings, which is what we recommend. However, if desired, all operations can be modified by adding options to the command line. For example, in Operation 1a, the user can change the number of SNPs used when testing for structure using the option "–num-pedigree-predictors", or skip this operation by adding "–check-pedigree NO". In Operation 1b, the user can change the numbers of random vectors and partitions using the options "–num-MCMC" and "–divide", respectively (note that the former will also change the number of random vectors used in Operation 1c). In Operation 1c, the user can change the maximum number of scans performed by the variational Bayes solver using the option "–num-scans" (note that this will also set the maximum number of scans used in Operations 1d & 1e). In Operation 1d, the user can change the proportion of test individuals using the option "–cv-proportion". In Operation 1e, the variational Bayes solver considers a chunk converged when the approximate log likelihood changes by less than $n \times tol$; by default, $tol = 10^{-6}$, but the user can change this value using the option "–tolerance". In Operation 1e, the user can change the number of SNPs used for the Grammar-Gamma formula using the option "–num-calibration-predictors" (as a

reminder, this formula is only used if the test in Operation 1a finds strong structure).

**Multi-phenotype version of LDAK-KVIK.** LDAK-KVIK can accommodate multiple phenotypes in Step 1, in which case it performs Operations 1b-1e once for each phenotype (it is not necessary to perform Operation 1a multiple times, because the test for structure does not depend on the phenotype). If a phenotype has missing values, then these are replaced by the observed mean for this phenotype (note that when running Step 2, the corresponding individuals will be excluded). In general, analyzing multiple phenotypes reduces runtime, relative to analyzing each phenotype separately. This is mainly because many of the operations perform matrix-matrix multiplications, which are more efficient when combined, and because analyzing multiple phenotypes has limited impact on the number of scans. For example, Supplementary Table 2 shows that it is almost four times faster to analyze ten quantitative phenotypes simultaneously than separately.

## 2.4 Binary Phenotypes

Suppose now the phenotype is binary, and let the length-$n$ vector $\mu$ contain the probabilities that each individual is a case (i.e., $\mu_i = P(Y_i' = 1)$). LDAK-KVIK obtains $\mu'$, an estimate of $\mu$, by regressing $Y'$ logistically on $Z$, then constructs $D$, an $(n \times n)$ diagonal matrix such that $D_{i,i} \propto \mu_i(1-\mu_i)$ and $Tr(D^{-1}) = n$.

LDAK-KVIK then performs the operations described above, with three general changes (see Supplementary Note 5 for a justification). Firstly, the standardized residual genotypes are obtained by setting $X = H'X'$, where $H' = I - Z(Z^T D Z)^{-1} D Z^T$, then scaling columns of $X$ to have variance one. Secondly, $Y$ contains standardized adjusted phenotypes, obtained by setting $Y = D^{-1}(Y - \mu')$, then scaling $Y$ to have variance one. Thirdly, whenever using a linear random-effects model of the form $Y = X\gamma + e$, LDAK-KVIK assumes that the covariance matrix of the noise term is proportional to $D^{-1}$ (i.e., the assumption $e \sim N(0, I(1-h^2))$ is replaced by $e \sim N(0, D^{-1}(1-h^2))$). The third change means that the variance matrices have the revised form $V = Kh^2 + D^{-1}(1-h^2)$, while Step 2a switches from ordinary least-squares regression to weighted least-squares regression (using $D$ as a weight matrix). Additionally, when performing Monte Carlo REML, the derivative ratio becomes

$$\delta(h^2) = \frac{Y^T V^{-1} K V^{-1} Y}{Y^T V^{-1} D^{-1} V^{-1} Y} \Big/ \frac{Tr(KV^{-1})}{Tr(D^{-1}V^{-1})}.$$

LDAK-KVIK makes two further changes when analyzing binary phenotypes. Firstly, for SNPs with association $p$-value below 0.1, it recomputes their test statistic using our novel SPA solver (i.e., replaces $\lambda U_j$ with $\lambda U_j'$, the SPA test statistic defined in the Online Methods). Secondly, when the test in Operation 1a determines there is strong structure, LDAK-KVIK switches to an approximate version of fastGWA (described in the Online Methods). The second change is motivated by the observations that when analyzing binary phenotypes, all MMAA tools tend to have very similar power, and that the choice of prior distribution for SNP effect sizes is less important when there is strong structure. Therefore, we find that switching to fastGWA has limited impact on power, but can substantially decrease runtime (e.g., when analyzing the 63 k individuals in the twins dataset, the CPU hours reduced from about 5 to 1).

## Supplementary Note 3: Using residual genotypes and phenotypes

Here we justify why our regressions use residual genotypes and phenotypes (or in the case of binary phenotypes, residual genotypes and adjusted phenotypes).

Suppose we regress the quantitative phenotype $Y'$ on $Z$ and $X'$ using ordinary least-squares regression. The estimated effect sizes are

$$\hat{\theta} = \begin{pmatrix} Z^T Z & Z^T X' \\ X'^T Z & X'^T X' \end{pmatrix}^{-1} \begin{pmatrix} Z^T Y' \\ X'^T Y' \end{pmatrix}.$$

We can evaluate the first term using a blockwise inversion. If we let $H = I - Z(Z^T Z)^{-1} Z^T$, then

$$\hat{\theta} = \begin{pmatrix} (Z^T Z)^{-1} + (Z^T Z)^{-1} Z^T X' (X'^T H X')^{-1} X'^T Z (Z^T Z)^{-1} & (Z^T Z)^{-1} Z^T X' (X'^T H X')^{-1} \\ -(X'^T H X')^{-1} X'^T Z (Z^T Z)^{-1} & -(X'^T H X')^{-1} \end{pmatrix} \begin{pmatrix} Z^T Y' \\ X'^T Y' \end{pmatrix}.$$

If $\theta_X$ denotes the sub-vector of $\theta$ corresponding to the genotypes, then its estimate is

$$\hat{\theta}_X = -(X'^T H X')^{-1} X'^T Z (Z^T Z)^{-1} Z^T Y' + (X'^T H X')^{-1} X'^T Y' = (X'^T H X')^{-1} X'^T H Y'.$$

The matrix $H$ is the complement of the hat matrix when performing ordinary least-squares regression with predictor matrix $Z$ (i.e., $HA$ is the residual of the matrix $A$ after regressing on $Z$). Further, $H$ is idempotent (i.e., $HH = H$). It follows that

$$\hat{\theta}_X = ((HX)^T (HX))^{-1} (HX)^T (HY') = (X^T X)^{-1} X^T Y.$$

Thus we can see that the estimated SNP effect sizes from regressing $Y'$ on $Z$ and $X'$ are identical to those from regressing $Y$ on $X$. The only complication with this approach is that when estimating the variance of the effect sizes, we must allow for the fact we have regressed out covariates (i.e., remember to divide by $n - q$ instead of $n - 1$).

Note that the above mathematics can be used to show that when performing weighted least-squares regression (with weight matrix $D$), the estimated effect sizes from regressing $Y'$ on $Z$ and $X'$ are identical to those from regressing $H'Y'$ on $H'X'$, where $H' = I - Z(Z^T DZ)^{-1} Z^T D$. However, when we perform weighted least-squares regression, we take a slightly different approach. Specifically, while we do use residual genotypes (i.e., set $X = H'X'$), we use adjusted phenotypes instead of residual phenotypes (i.e., set $Y = D^{-1}(Y' - \mu)$ instead of $Y = H'Y'$). This reflects that we use weighted least-squares regression as an approximation to logistic regression (as explained in Supplementary Note 5). Furthermore, it can be shown that when using residual genotypes and adjusted phenotypes, the resulting score test statistic matches that obtained from regressing $Y'$ logistically on $X'$ and $Z$ (see Appendix A of Dey *et al.* [30] for a proof). However, we recognise that, in practice, the impact of replacing residual phenotypes with adjusted phenotypes is likely to be small (essentially, we are replacing residual phenotypes from weighted linear regression with weighted residual phenotypes from logistic regression).

## Supplementary Note 4: Expected residuals for the Ridge Regression PRS.

Let $V = Kh^2 + I(1 - h^2)$, with $K = X\Omega X^T$. Here we prove the statement in the Online Methods that when constructing a Ridge Regression PRS $Y = X\hat{\gamma}$, assuming

$$Y = X\gamma + e, \quad \text{with} \quad \gamma_j \sim N(0, \Omega_{j,j} h^2) \quad \text{and} \quad e_i \sim N(0, 1 - h^2),$$

then the PRS takes the form $P = h^2 K V^{-1} Y$ and the corresponding residuals are $Y - P = (1-h^2)V^{-1}Y$.

With the priors defined above, the posterior distribution of SNP effect sizes satisfies

$$P(\gamma|Y) \propto \exp\left(-\frac{(Y-X\gamma)^T(Y-X\gamma)}{2(1-h^2)}\right) \times \exp\left(-\frac{\gamma^T\Omega^{-1}\gamma}{2h^2}\right),$$

which can be rewritten as

$$P(\gamma|Y) \propto \exp\left(-\frac{(\gamma-A)^T B^{-1}(\gamma-A)}{2}\right) \quad \text{where} \quad B^{-1} = \frac{X^TX}{1-h^2} + \frac{\Omega^{-1}}{h^2} \quad \text{and} \quad A = (X^TX + \frac{1-h^2}{h^2}\Omega^{-1})^{-1}X^TY.$$

Therefore, the posterior distribution of $\gamma$ is multivariate normal, with mean $A$ and variance $B$. By applying the Woodbury matrix identify, we can write

$$A = \left(\frac{h^2}{1-h^2}\Omega - \frac{h^2}{1-h^2}\Omega X^T\left(\frac{h^2}{1-h^2}X\Omega X^T + I\right)^{-1} X\frac{h^2}{1-h^2}\Omega\right)X^TY$$

Using the definitions of $V$ and $K$ from above, it follows that the PRS constructed using the posterior mean of $\gamma$ is

$$P = X\hat{\gamma} = XA = \frac{h^2}{1-h^2}K\left[I - \left(\frac{V}{1-h^2}\right)^{-1}\frac{h^2}{1-h^2}K\right]Y = h^2KV^{-1}Y,$$

and the corresponding vector of residuals is

$$Y - P = \left(I - h^2KV^{-1}\right)Y = (1-h^2)V^{-1}Y.$$

Note that when the phenotype is binary, we instead obtain

$$P = h^2KV^{-1}Y \quad \text{and} \quad Y - P = (1-h^2)D^{-1}V^{-1}Y,$$

where the variance matrix now takes the form $V = Kh^2 + D^{-1}(1-h^2)$.

## Supplementary Note 5: Approximate logistic regression

In this section, we assume $Y'$ is a binary phenotype, and explain our reason for switching from ordinary least-squares regression to weighted least-squares regression. As a reminder, we use the length-$n$ vector $\mu$ to denote the probabilities that each individual is a case, while $\mu'$ is an estimate of $\mu$ from regressing $Y'$ logistically on $Z$ (i.e., $\mu'_i$ is the estimated probability that Individual $i$ is a case given their covariates).

Suppose we were to regress $Y'$ logistically on the matrices $Z$ and $X'$ using the model $\log(\mu/(1-\mu)) = A\theta$, where the matrix $A$ contains both the covariates and the genotypes (i.e., $A = [Z\ X']$). The corresponding log likelihood would be

$$l(Y'|A) = \sum_i \big(Y_i \log(\mu_i) + (1-Y_i)\log(1-\mu_i)\big),$$

whose first and second derivatives can be expressed as

$$\frac{dl}{d\theta} = A^T(Y'-\mu) \quad \text{and} \quad \frac{d^2l}{d\theta^2} = -A^TD'A,$$

where $D'$ is a diagonal matrix with $D'_{i,i} = \mu_i(1-\mu_i)$. We could estimate $\theta$ using Newton-Raphson iterations of the form

$$\theta^{k+1} = \theta^k + (A^TD'A)^{-1}A^T(Y'-\mu),$$

where $\theta^k$ denotes the estimated coefficients after the $k$th iteration. If we rewrite this as

$$\theta^{k+1} = \theta^k + (A^T D' A)^{-1} A^T D' [D'^{-1}(Y' - \mu)],$$

then we can see that the updated estimate of $\theta$ is similar to that obtained by regressing the adjusted phenotype $D'^{-1}(Y' - \mu)$ linearly on $A$ with weight matrix $D'$ (in fact, the two estimates are equal when $\theta^k = 0$).

Now let $\theta_Z$ and $\theta_X$ denote the sub-vectors of $\theta$ corresponding to the covariates and genotypes, respectively. Further, suppose $\theta_Z^0$, the starting values for $\theta_Z$, are set to their values from regressing $Y'$ logistically on $Z$, while $\theta_X^0$, the starting values for $\theta_X$, are set to zero. It can be shown that the estimate of $\theta_X$ after one Newton-Raphson iteration is

$$\theta_X^1 = (X_j^T D X_j)^{-1} X_j^T D Y,$$

where $D$ is the realization of the weight matrix when $\mu = \mu'$, $X$ contains the residuals from regressing columns of $X'$ linearly on $Z$ with weight matrix $D$, and $Y = D^{-1}(Y' - \mu)$.

Therefore, our decision to switch to weighted least-squares regression when the phenotype is binary, is because the resulting estimates of SNP effect sizes can be considered an approximation of those obtained from logistic regression. We note that we could improve the accuracy of this approximation by updating the weight matrix $D$ during the algorithm (instead of fixing its value at the start), in which case our approach would mirror the quasi-likelihood approach used by SAIGE [20] and fastGWA [7]. However, updating $D$ would introduce computational challenges (e.g., there would need to be a separate $D$ for each PRS, which would prevent us from computing terms such as $X^T D(Y - P)$ for multiple PRS at once). Furtermore, we believe the benefit of updating $D$ would be small. This is because the major advantage of switching from linear to logistic regression is when there are relatively few cases, but in these circumstances, the proportion of variance explained by the SNPs tends to be small, and therefore the updated $D$ would be relatively close to its starting value. This is evidenced by the results in Supplementary Figure 9, which show that our approximate version of fastGWA (which does not update $D$) has well-controlled type 1 error.

# Supplementary Note 6: Data

We use genotypic and phenotypic data from the UK Biobank [23, 24] (obtained via application 21432).

## 6.1 Genotypes

After excluding individuals missing phenotypic information, or those who have withdrawn consent, the UK Biobank contains approximately 487 k individuals, directly genotyped for approximately 784 k SNPs. We constructed four datasets: the "white dataset" contains 367 981 individuals and 690 264 SNPs, the "homogeneous dataset" and the "twins dataset" both contain 63 000 individuals and 690 264 SNPs, while the "multi-ancestry dataset" contains 60 019 individuals and 471 760 SNPs. We primarily used the white dataset when analysing the real UK Biobank phenotypes, and the three remaining datasets for simulations.

To create the white dataset, we first restricted to the 408 868 individuals listed in data field 22006 (those who self-identify as being white British, and whom UK Biobank determined "have very similar genetic ancestry based on

a principal components analysis of the genotypes"). We then identified 690 264 autosomal, biallelic SNPs with MAF $> 0.001$ and genotype call rate $> 90\%$. Finally, we randomly sampled 367 981 (90%) of the individuals (we use the remaining 40 887 individuals for testing the accuracy of Step 1 PRS).

We constructed the homogeneous dataset by sampling 63 000 distantly-related individuals from the white dataset (when sampling individuals, we ensured there were no first or second degree relatives, based on relationships inferred by the software KING [33]). We constructed the twins dataset by randomly picking 31 500 individuals from the homogeneous dataset and duplicating their genomes. We constructed the multi-ancestry dataset by combining 8 751 individuals from the white dataset with 51 268 individuals who did not identify as being white British.

Supplementary Figure 25 provides principal component plots for the UK Biobank data, while Supplementary Table 4 gives an overview of the ethnic composition of each dataset. Note that when analyzing simulated phenotypes, we included the top ten principal component axes as covariates, while when analyzing real phenotypes, we included the top ten axes, and also age, sex, the square of age, and age times sex.

## 6.2 Simulated phenotypes

Note that in this section, there is a slight abuse of notation, because we use the index $j$ to denote the $j$th causal SNP (instead of the $j$ SNP overall).

We first partitioned the genome into three: the "causal partition" contains the starts of each chromosomes, the "buffer partition" contains the middle 4 Mb of each chromosome, while the "null partition" contains the ends of each chromosome (see Supplemental Figure 3). When simulating quantitative phenotypes, we specified $m'$, the number of causal SNPs (picked at random from the causal partition), $h^2$, the heritability contributed by all causal SNPs, and the power parameter $\alpha$. When simulating binary phenotypes, we also specified $K$, the prevalence. Note that the choices of $m'$, $h^2$ and $\alpha$ determine the values of $\mathbb{E}[h_j^2]$, the expected heritability contributed by the $j$th causal SNP.

We generated each quantitative phenotype using the model

$$Y = \sum_{j=1}^{m'} X_j \gamma_j + A\theta_1 + e$$

where $X_j$ and $\gamma_j$ denote the genotypes and effect size of the $j$th causal SNP, respectively, $A$ is a length-$n$ vector containing the first principal component axes, $\theta_1$ is a scalar, while $e$ is a length-$n$ vector representing environmental noise. We sampled $\gamma_j$ and $e_i$ independently from the mean-zero normal distributions $\gamma_j \sim N(0, \hat{h}_j^2)$ and $e_i \sim N(0, 1 - h^2)$, respectively, which ensures that the expected heritability of the $j$th causal SNP equals $\mathbb{E}[h_j^2]$, then we set $\theta_1$ so that the first principal component explained 5% of total phenotypic variance.

For binary phenotypes, we first simulated liabilities using the model

$$L = \sum_{j=1}^{m'} X_j \gamma_j + A\theta_1 + e$$

with $\gamma_j$, $\theta_1$ and $e_i$ generated the same as for quantitative phenotypes. We then obtained phenotypes by dichotomizing

the liabilities. Specifically, having decided the prevalence $K$, we set $Y_i = 1$ if $L_i > \Phi(K)^{-1}$, and $Y_i = 0$ if $L_i \leq \Phi(K)^{-1}$, where $\Phi^{-1}$ is the inverse CDF of a standard normal distribution.

### 6.3 Real phenotypes

Details of the 40 quantitative and 20 binary UK Biobank phenotypes are provided in Supplementary Tables 5 & 6. Note that across the 368 k individuals in the white dataset, the quantitative phenotypes have on average 5.1% missing values (range 0.2% to 13.6%), whereas the binary phenotypes are complete (because everyone not identified as being a case, was automatically assumed to be a control).

## Supplementary Note 7: Existing MMAA Tools

We benchmark the performance of LDAK-KVIK against classical regression and four existing MMAA tools: BOLT-LMM [1], REGENIE [6], fastGWA [7] and GCTA-LOCO [8]. Note that REGENIE and fastGWA are designed for both quantitative and binary phenotypes, whereas BOLT-LMM and GCTA-LOCO are designed for quantitative phenotypes. Here we provide a summary of each tool. Note that all four existing MMAA tools (implicitly) assume that $\mathbb{E}[h_j^2]$ is constant (equivalent to setting $\alpha = -1$ in Equation 2).

**BOLT-LMM.** BOLT-LMM [1] is a two-step MMAA tool. In Step 1, it creates LOCO Mixture-Normal PRS, where the SNP effect sizes are assigned the following prior distribution

$$\gamma_j \sim pN\left(0, \frac{1-F}{p} \times \frac{\hat{h}^2}{m}\right) + (1-p)N\left(0, \frac{F}{1-p} \times \frac{\hat{h}^2}{m}\right).$$

BOLT-LMM estimates $h^2$ using Monte Carlo REML, and determines suitable values for $p$ and $F$ via cross-validation. BOLT-LMM estimates the test statistic scaling factor $\lambda$ using LDSC [19] (as explained in Supplementary Note 2).

We find that BOLT-LMM is generally the most powerful existing MMAA tool for quantitative phenotypes, reflecting the benefits of using a mixture prior distribution for SNP effect sizes when constructing PRS. However, we also find that BOLT-LMM is computationally demanding, due to its genome-wide variational Bayes solver, and its use of conjugate gradient descent (both when performing Monte Carlo REML and when calculating the test statistic scaling factor).

We have shown that BOLT-LMM can overestimate $\lambda$, which leads to inflated test statistics (this is mainly because its estimates of $\lambda$ are derived from LDSC, which assumes $\alpha = -1$). We therefore created BOLT-LMM-Unscaled, whose results are those that BOLT-LMM would produce if it was forced to estimate $\lambda = 1$ (i.e., having run BOLT-LMM, we divide its reported $\chi^2(1)$ test statistics by its estimate of $\lambda$).

When running BOLT-LMM, we used the option "–lmmForceNonInf" to ensure it always computes test statistics using the LOCO Mixture-Normal PRS as offsets, and we used the option "—LDscoresUseChip" so that it computes LD scores from the data. Further, we used the option "–predBetasFile" to save the estimated effect sizes of the Bolt-LMM PRS (this allowed us to measure the accuracy of the BOLT-LMM PRS when applied to independent

data, and compare this to the accuracy of the LDAK-KVIK PRS).

**REGENIE.** REGENIE [6] is a two-step MMAA tool. In Step 1, it creates PRS using stacked ridge regression. First, it divides the genome into blocks, then for each block it creates five PRS assuming the effect size prior distribution $\gamma_j \sim N(h^2/m)$, with $h^2$ set to 0.01, 0.25, 0.5, 0.75 or 0.99. Finally, it uses ridge regression to combine all block-based PRS into LOCO PRS. REGENIE does not compute a test statistic scaling factor (so in effect, always assumes $\lambda = 1$).

REGENIE is computationally efficient, reflecting that it only needs to store one block of SNPs at a time, and that the block-based PRS are fast to compute and combine. However, we find that REGENIE is less powerful than BOLT-LMM, due to the infinitesimal nature of its prior distribution for SNP effect sizes.

When running REGENIE, we set the block-size to 1000 SNPs. When analyzing binary phenotypes, REGENIE recomputes small $p$-values using either its implementation of fastSPA [30], or using an approximate Firth correction [34]; we followed the recommendation of the authors and selected the latter.

**fastGWA.** A description of fastGWA is provided in Supplementary Note 2. While our applications of fastGWA demonstrated its computational efficiency, they also found that it tended to be the least-powerful MMAA tool. The latter reflects that a sparse variance matrix will poorly capture the contributions of SNPs distant from the SNP being tested (consider the extreme case, where all off-diagonal terms in $K$ are set to zero, in which case the model used by fastGWA would be almost identical to that used by classical linear regression). Furthermore, restricting to a sparse variance matrix means that fastGWA can only account for heterogeneity due to familial relatedness, but not due to population structure (for this reason, we did not use fastGWA to analyze the multi-ancestry dataset).

**GCTA-LOCO.** GCTA-LOCO [8] performs generalized least-squares regression using the model $Y \sim N(Z\theta + X_j\beta_j, \hat{V}_c)$, where $\hat{V}_c$ is an estimated LOCO variance matrix taking the form $V = Kh^2 + I(1-h^2)$. Unlike fastGWA, GCTA-LOCO does not require the variance matrices to be sparse. Therefore, GCTA-LOCO tends to be more powerful than fastGWA (because it better captures the contributions of SNPs distant from the SNP being tested), but also much more computationally-demanding. However, GCTA-LOCO tends to be less powerful than BOLT-LMM, reflecting that its model arises from assuming an infinitesimal prior distribution for SNP effect sizes.

## Supplementary Note 8: Our attempts to estimate $\lambda$

As explained in Supplementary Note 2, the Grammar-Gamma Formula provides a way to estimate $\lambda$ when performing two-step MMAA using Ridge Regression PRS. [18] However, we are not aware of a corresponding formula for the general case (e.g., when using Elastic Net PRS). Although BOLT-LMM proposed estimating $\lambda$ based on LDSC, our simulations demonstrate that this approach is unreliable, and can result in substantially inflated test statistics. Here we summarize the seven main strategies we considered, before opting for the approach described in Supplementary Note 2 (i.e., using a test for structure to decide whether it is valid to set $\lambda = 1$, else switching from a Elastic Net PRS to a Ridge Regression PRS).

Note that in the following paragraphs, we use $T_j$ and $U_j$ to denote $\chi^2(1)$ test statistics obtained by regressing $Y - P_c$ on $X_j$, where $P_c$ is either a LOCO Ridge Regression PRS ($T_j$) or a LOCO Elastic Net PRS ($U_j$). Further, we use $\lambda'$ to denote the estimated test statistic scaling factor corresponding to $T_j$ (obtained using the Grammar-Gamma Formula).

**1 - Estimating $\lambda$ using SumHer.** We tried modifying the approach proposed by BOLT-LMM, so that instead of measuring confounding using LDSC (which assumes $\alpha = -1$), we used our software SumHer (which allows the user to specify $\alpha$). [35] We therefore ran SumHer twice, first using the scaled test statistics $\lambda' T_1, \ldots, \lambda' T_m$, then using the unscaled test statistics $U_1, \ldots, U_m$. For both runs, we used the LDAK-KVIK estimate of $\alpha$ (obtained in Operation 1b using randomized Haseman-Elston Regression). Finally, we set $\lambda = A_1/A_2$, where $A_1$ and $A_2$ were the estimated intercepts from the two runs of SumHer.

Although we found improvement using SumHer instead of LDSC (in particular, it greatly reduced the inflation of test statistics when analyzing phenotypes generated assuming $\alpha = -0.25$), our modified approach still appeared to produce biased test statistics. Additionally, we had three further concerns with this approach. Firstly, the estimates of confounding (both from LDSC and SumHer) are based on strong assumptions regarding how causal variation is distributed across the genome and the impact of confounding on test statistics, but it is challenging to test the validity of these assumptions on real data. Secondly, this approach can only be applied to datasets with linkage disequilibrium (i.e., where nearby predictors tend to be strongly correlated), whereas we wanted an approach that could be universally applied. Thirdly, this approach requires computation of two sets of PRS, which leads to an increased runtime.

**2 - Estimating $\lambda$ based on the test statistics of weakly-associated SNPs.** This approach was similar to Approach 1, except that we measured confounding based on the average test statistic of SNPs that showed only weak association with the phenotype. Specifically, we set $\lambda$ so that $\sum_{j \in \mathbb{S}} \lambda' T_j = \sum_{j \in \mathbb{S}} \lambda U_j$, where the set $\mathbb{S}$ contained the indexes of "weakly-associated SNPs". We tried multiple ways to define weakly-associated SNPs. For example, we considered SNPs with $\lambda' T_j < 2$, or SNPs with $\lambda' T_j < 1$, or SNPs with both $\lambda' T_j < 1$ and $\lambda U_j < 1$ (the latter definition sought to avoid biases caused by Winners' Curse, and required us to iteratively update $\mathbb{S}$ and $\lambda$). In general this approach showed promise, however, we found that it tended to produce slightly deflated test statistics (i.e., appeared to underestimate $\lambda$), was sensitive to the criterium used to define weakly-associated SNPs, and required the computation of two sets of PRS.

**3 - Estimating $\lambda$ based on the test statistics of strongly-associated SNPs.** When using a two-step MMAA tool, the expected increase of the average $\chi^2(1)$ test statistic of associated SNPs is proportional to 1/MSE, where MSE is the average mean-squared error of the Step 1 LOCO PRS. This motivated us to set $\lambda$ so that $\sum_{j \in \mathbb{S}} \lambda U_j / \sum_{j \in \mathbb{S}} \lambda' T_j = M_1/M_2$, where the set $\mathbb{S}$ contained the indexes of strongly-associated SNPs (e.g., SNPs with $\lambda' T_j > 30$), while $M_1$ and $M_2$ are, respectively, the estimated MSE of the LOCO Ridge Regression and Elastic Net PRS (obtained from the cross-validation performed in Operation 1d). We found this approach performed poorly, reflecting the large variance of test statistics (e.g., the standard deviations of test statistics with expectation 30 and 50 are approximately 11 and 14, respectively).

**4 - Generalized least-squares regression with mixture priors** Suppose we have only two SNPs, and test SNP 1 for association using the model $Y = X_1\beta_j + X_2\gamma_2 + e$, with $\gamma_2 \sim N(0, \hat{h}^2)$ and $e_i \sim N(0, 1-\hat{h}^2)$. This would be equivalent to assuming the model

$$Y \sim N(X_1\beta_1, \hat{V}) \quad \text{with} \quad \hat{V} = K\hat{h}^2 + I(1-\hat{h}^2) \quad \text{where} \quad K = X_2X_2^T.$$

The generalized least-squares estimate of $\beta_1$ is

$$\hat{\beta}_1 = \frac{X_1^T\hat{V}^{-1}Y}{X_1^T\hat{V}^{-1}X_1},$$

which is the value of $\beta_1$ that maximizes the model likelihood

$$L(Y|\beta_1) = (2\pi)^{\frac{-(n-q)}{2}}|\hat{V}|^{\frac{-1}{2}}\exp\left(-\frac{(Y-X_1\beta_1)^T\hat{V}^{-1}(Y-X_1\beta_1)}{2}\right).$$

Now suppose we replace the infinitesimal prior distribution for $\gamma_2$ with the mixture prior distribution $\gamma_2 \sim pN(0, h_A^2) + (1-p)N(0, h_A^2)$. This would be equivalent to assuming the model

$$Y \sim pN(X_1\beta_1, \hat{V}_A) + (1-p)N(X_1\beta_1, \hat{V}_B), \quad \text{with} \quad \hat{V}_A = Kh_A^2 + I(1-h^2) \quad \text{and} \quad \hat{V}_B = Kh_B^2 + I(1-h^2),$$

and the corresponding likelihood takes the form

$$L(Y|\beta_1) = pL_A(Y|\beta_1) + (1-p)L_B(Y|\beta_1),$$

where $L_A(Y|\beta_1)$ and $L_B(Y|\beta_1)$ are the likelihoods when assuming $Y \sim N(X_1\beta_1, \hat{V}_A)$ and $Y \sim N(X_1\beta_1, \hat{V}_B)$, respectively. One way to estimate $\beta_1$ would be to maximize the revised model likelihood, which could be considered the generalized least-squares estimate assuming a mixture prior. If we did this directly, by differentiating the model likelihood with respect to $\beta_1$ and setting to zero, it would be necessary to solve

$$-X_1^T\hat{V}_A^{-1}(Y-X_1\beta_1) \times pL_A(Y|\beta_1) - X_1^T\hat{V}_B^{-1}(Y-X_1\beta_1) \times (1-p)L_B(Y|\beta_1) = 0.$$

It is not obvious how to find $\beta_1$. Furthermore, consider that this is the simplest case, because there are only two SNPs, and hence only one $\gamma_j$; for a proper application, there would be approximately $m$ $\gamma_j$, and the model likelihood (and its derivative) would have approximately $2^m$ terms. We wondered whether it would be feasible to replace the (true) model likelihood with the approximate likelihood from variational Bayes. However, by our calculations, this would be equivalent to assuming $Y - P_c \sim N(X_1\beta_1, I(1-h^2))$, and therefore very similar to performing classical linear regression.

**5 - Finding an equivalent Ridge Regression PRS.** As explained in Supplementary Note 2, a Ridge Regression PRS is obtained by assuming prior distributions of the form $\gamma_j \sim N(0, a_j)$ and $e_i \sim N(0, \sigma_e^2)$. We noticed that, given an arbitrary LOCO Elastic Net PRS $P_c$, our variational Bayes solver could find values for $a_j$ and $\sigma_e^2$ such that the resulting Ridge Regression PRS $P_c'$ was identical to $P_c$. It followed that if $\Omega'$ is an $m \times m$ diagonal matrix with $\Omega'_{j,j} = a_j$, then we could express each LOCO Elastic Net PRS constructed in Step 1 in the form $P_c = K'V'^{-1}Y$, where $K' = X\Omega'X^T$ and $V' = K' + I\sigma_e^2$. This suggested that we could use the Grammar-Gamma Formula, but replacing $V$ with $V'$. Unfortunately, this approach did not work, which we suspect was due to overfitting, a consequence of the fact that the matrix $V'$ was constructed based on the data.

**6 - Estimating $\lambda$ via permutation.** The role of $\lambda$ can be better understood by examining the per-SNP components of the Grammar-Gamma Formula

$$\lambda'_j = \frac{X_j^T X_j \times \hat{s}^2}{X_j^T \hat{V}_c^{-1} X_j \times \hat{\sigma}^2}, \quad \text{where} \quad \hat{s}^2 = \frac{Y^T \hat{V}_c^{-1} \hat{V}_c^{-1} Y}{n-q} \quad \text{and} \quad \hat{\sigma}^2 = \frac{Y^T \hat{V}_c^{-1} Y}{n-q}.$$

Note that this version is slightly different to that in Supplementary Note 2, because we previously ignored the LOCO aspect.

We believe that $\lambda$ has two main purposes. The first is to quantify the overfitting due to correlation between SNPs on different chromosomes, while the second is to quantify the the overfitting inherent in the LOCO PRS. Both these purposes are captured by the denominator term $X_j^T V_c^{-1} X_j$. To appreciate why, note that $V_c^{-1} X_j$ is proportional to the vector of residuals when constructing a Ridge Regression PRS for $X_j$ using SNPs on other chromosomes (Supplementary Note 4), and therefore its magnitude is a measure of overfitting.

We therefore considered computing $\lambda_j = G_j^T V'^{-1} G_j / X_j^T V'^{-1} X_j$, where $G_j$ is a permutation of $X_j$ and $V'$ is the variance matrix corresponding to the "equivalent Ridge Regression PRS" (explained in Approach 5). Specifically, our proposed estimate of $\lambda$ was the mean value of $\lambda_j$ computed for 20 randomly-picked SNPs. Our idea was that the numerator $X_j^T V'^{-1} X_j$ will be affected by both overfitting due to cross-chromosome correlations and inherent overfitting, whereas the denominator $G_j^T V'^{-1} G_j$ will only be affected by inherent overfitting (because permuting SNP $j$ will ensure this SNP is uncorrelated with SNPs on other chromosomes). It follows that the ratio of these two terms will measure only overfitting due to cross-chromosome correlations.

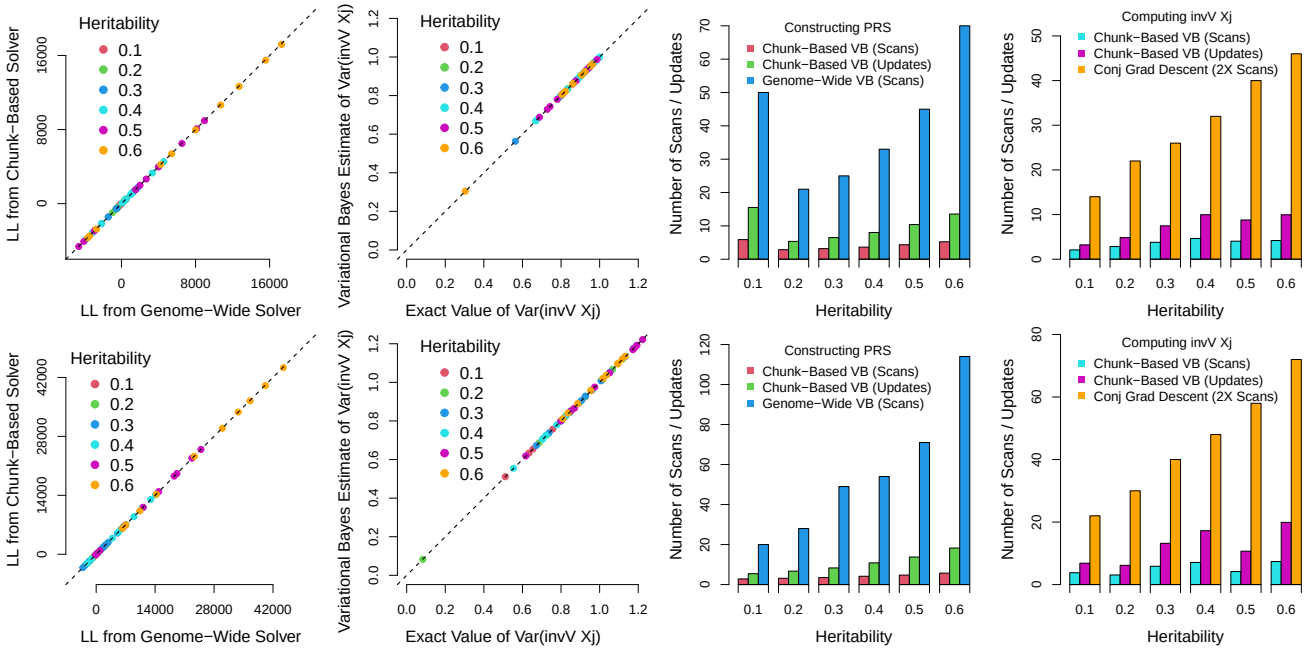
We found that this approach performed well for the homogeneous dataset (in which case cross-chromosome correlations are slight, and $\lambda_j$ is close to one). However, it only partially worked for the twins and multi-ancestry dataset (in general, we found it underestimated $\lambda$ by 10-20%).

**7 - Set $\lambda = 1$.** As explained above, we view $\lambda$ as a measure of overfitting, and therefore we believe its true value is upper-bounded by one (because we do not see how negative overfitting is possible). Therefore, we considered simply setting $\lambda = 1$. Although this can lead to deflated test statistics (when the true value is below one), it should not lead to inflated test statistics (because the true value is never above one). As demonstrated in Supplementary Figures 19 & 20, we found situations where this could increase power, relative to our chosen solution (i.e., switching to Ridge Regression PRS if strong structure was detected, then using the Grammar-Gamma Formula). This is because there are scenarios where the increased power from using Elastic Net PRS outweighs the reduced power caused by overestimating $\lambda$. However, we opted for our chosen solution for two reasons. Firstly, we considered these scenarios unusual (i.e., they require datasets containing a very specific amount of structure and phenotypes with relatively low polygenicity). Secondly, we believe that it is beneficial to have well-calibrated test statistics (i.e., those produced using Ridge Regression PRS and the Grammar-Gamma Formula), whereas those produced using Elastic Net PRS and setting $\lambda = 1$ would tend to be deflated.


**Supplementary Figure 1: Illustration of our chunk-based variational Bayes solver.**

We use variational Bayes to obtain $Q(\gamma) = \prod Q_j(\gamma_j)$, an approximation of the posterior distribution of SNP effect sizes. The variational Bayes solver contained within BOLT-LMM [1] uses sequential genome-wide scans. On Scan 1, it visits each SNP once (i.e., update $Q_1(\gamma_1)$, then $Q_2(\gamma_2)$, and so on, until $Q_m(\gamma_m)$). Then it repeats this process on subsequent scans (i.e., visits each SNP once), continuing under the (genome-wide) approximate log likelihood has converged. By contrast, our solver divides the genome into chunks of 256 SNPs, then uses chunk-based scans. On Scan 1, it first only updates $Q_j(\gamma_j)$ for SNPs in Chunk 1, continuing until the (chunk-based) approximate log likelihood for Chunk 1 has converged. It then moves on to SNPs in Chunk 2 (continuing until convergence), then to SNPs in Chunk 3, and so on until it reaches the final chunk in the genome. On subsequent scans, our solver only visits chunks that had a sizeable impact on the approximate log likelihood in the previous scan.

This figure shows a scenario where there are 2697 chunks. On Scan 1 our solver visits all chunks, and the same for Scans 2 & 3. However, on Scan 4, it only visits 1985 chunks, indicating that the remaining 712 achieved convergence on Scan 3. We see that by the end of Scan 8, all chunks have converged, so our solver stops. Within each scan, the height of each chunk indicates the number of updates required until each convergence, while the numbers in the right column record the average number of updates across all active chunks. For example, we see that on Scan 5, the average number of updates was 1.2 (i.e., for the approximately 153 k SNPs contained within the 600 chunks visited on this scan, it was on average necessary to update $Q_j(\gamma_j)$ 1.2 times).

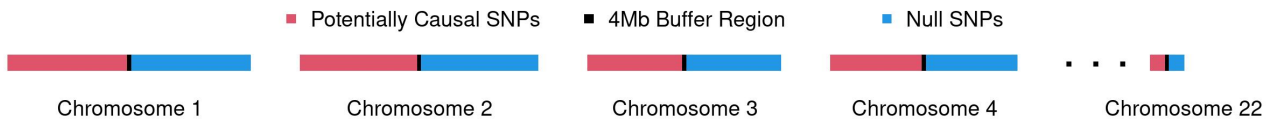



**Supplementary Figure 2: Accuracy and speed of our variational Bayes solver.**

For this figure, we generate six phenotypes each with 10 000 causal SNPs, and with heritability ranging from 0.1 to 0.6. The top panels report results from analyzing 50 k individuals, while the bottom panels report results from analyzing 100 k individuals. The first column shows that when constructing Elastic Net PRS, the approximate log likelihood from our chunk-based variational Bayes solver matches very closely that from a conventional, genome-wide solver. Note that for each phenotype, there are 10 points, corresponding to ten different sets of elastic net hyperparameters. Next we compute $V^{-1}X_j$ for twenty randomly-picked SNPs, where $V$ is the (phenotype-specific) estimated variance matrix. The second column shows that the variances of the estimates of $V^{-1}X_j$ from our variational Bayes solver are very close to the true values (which we computed using conjugate gradient descent).

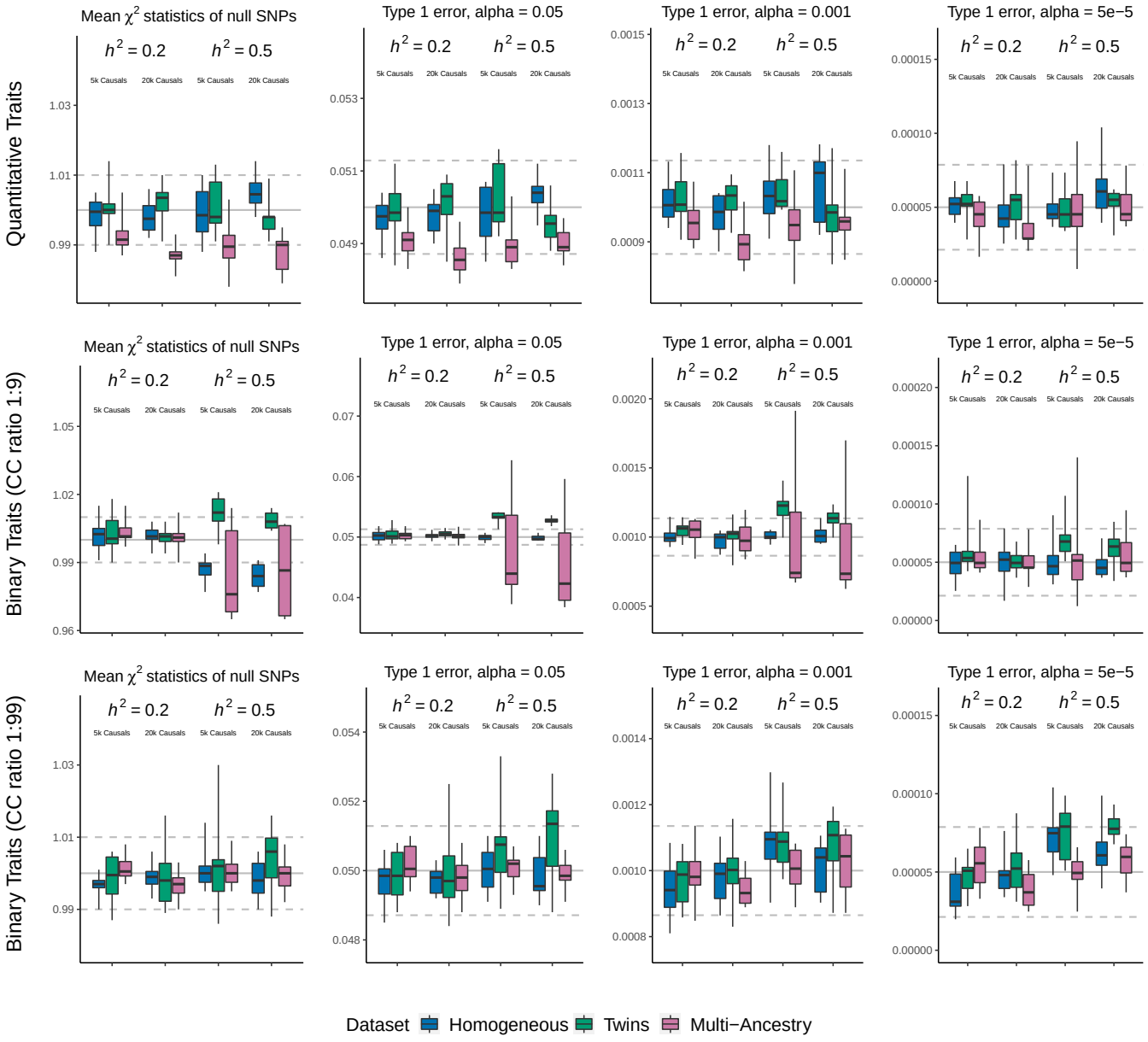
The third column reports the numbers of scans and per-SNP updates required by our chunk-based variational Bayes solver to construct the Elastic Net PRS for each phenotype, as well as the number of scans required by a genome-wide variational Bayes solver. For example, the first bar in the top panel shows that when analyzing the phenotype with heritability 0.1, our solver requires 5.9 scans. Note that this is not an integer, because our solver can perform partial scans (e.g., it may have been the case that our solver performed five scans that visited all chunks, then two scans that visited only 45% of chunks). The second bar shows that our solver performed 15.5 per-SNP updates (i.e., it updated $Q_j(\gamma_j)$ on average 15.5 times for each SNP). The third bar shows that the genome-wide solver required 50 scans (as each scan updates $Q_j(\gamma_j)$ once per SNP, this means the genome-wide solver required about three times more updates than our chunk-based solver). The fourth column is similar to the third, except it reports the numbers of scans and per-SNP updates our solver required when computing $V^{-1}X_j$, and compares this to twice the number of scans required when using conjugate gradient descent (we report twice the number of scans because each conjugate gradient descent scan performs approximately twice the number of algebraic operations as each scan of our variational Bayes solver).

In summary, we found that our chunk-based variational Bayes solver constructs PRS that match very closely those computed by a genome-wide variational Bayes solver, but requires substantially fewer updates of $Q_j(\gamma_j)$. We additionally found that our solver produces accurate estimates of $V^{-1}X_j$, and that it does so substantially faster than coordinate gradient descent.




**Supplementary Figure 3: Simulating phenotypes.**

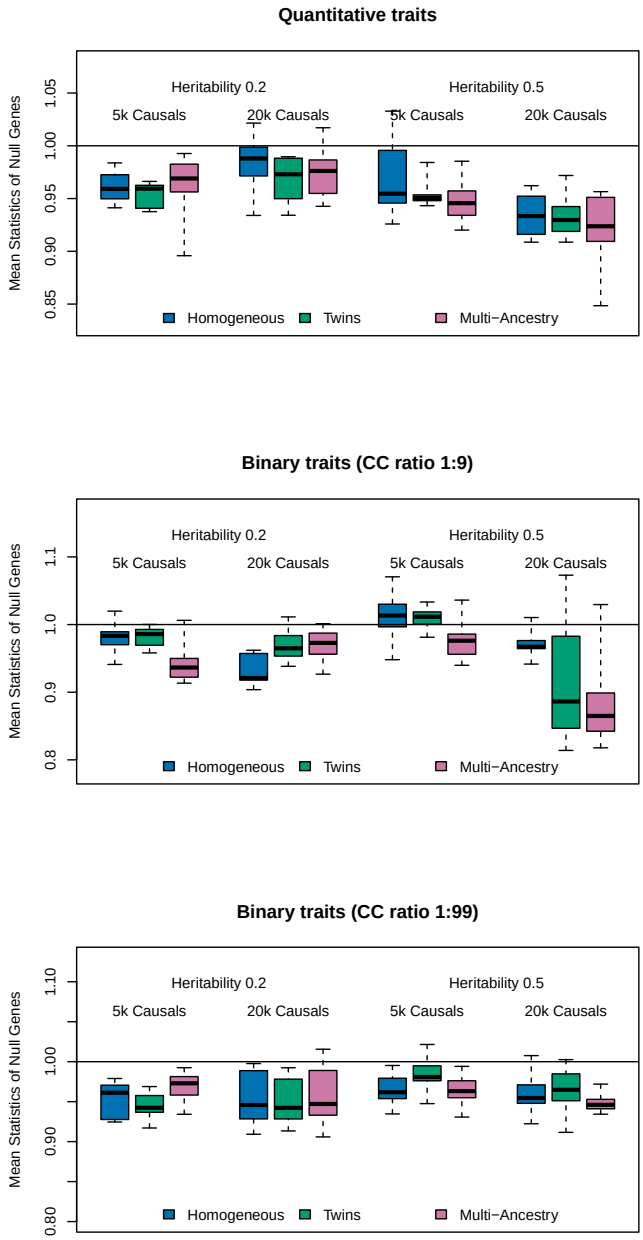
We divided the genome into three partitions, as shown above. SNPs within the central 4 Mb of each chromosome were assigned to the black partition, while the SNPs upstream and downstream were assigned to the red and blue partitions, respectively. When simulating phenotypes, we ensured that all causal SNPs were located within the red partition, then used the SNPs within the blue partition for measuring type 1 error (i.e., these were considered "null SNPs"). We included the black partition as a buffer, to ensure the null and causal SNPs were in approximate linkage equilibrium.




**Supplementary Figure 4: Type 1 error of LDAK-KVIK.**

We simulate quantitative and binary phenotypes for the homogeneous, twins and multi-ancestry datasets (which contain 63 k, 63 k and 60 k individuals, respectively). For each dataset, we consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using LDAK-KVIK, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null snps (Column 1), and based on the proportion of null SNPs with $p$-value below 0.05, 0.001 or $5 \times 10^{-5}$ (Columns 2, 3 & 4, respectively). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected value of each measure under the null hypothesis, while the dashed grey lines provide a 95% confidence interval (derived by analyzing 200 permuted phenotypes).

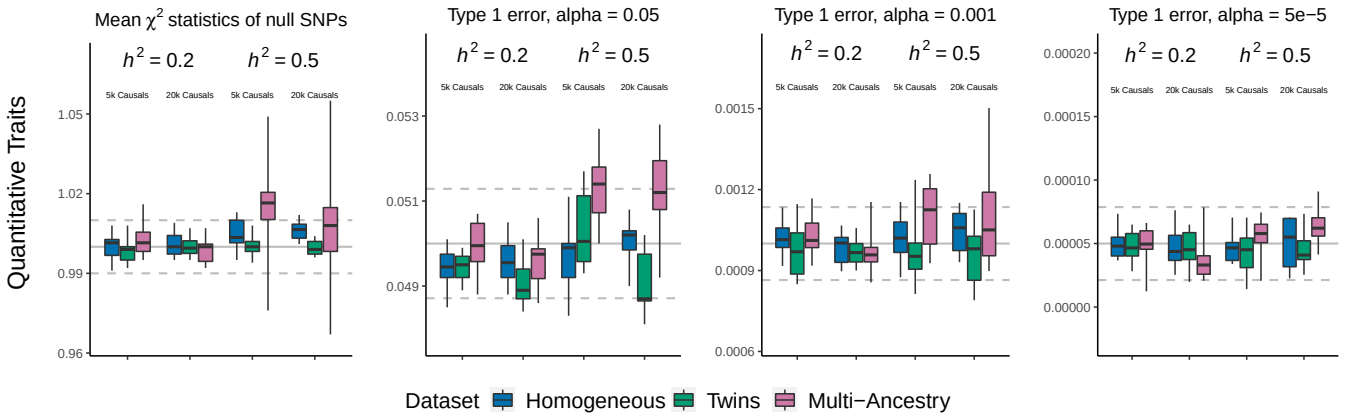
We find that LDAK-KVIK has well-controlled type 1 error for all datasets and scenarios considered.




**Supplementary Figure 5: Type 1 error of LDAK-KVIK-GBAT.**

We simulate quantitative and binary phenotypes for the homogeneous, twins and multi-ancestry datasets (which contain 63 k, 63 k and 60 k individuals, respectively). For each dataset, we consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform gene-based analysis using LDAK-KVIK-GBAT, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null genes. In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected mean $\chi^2(1)$ test statistic under the null hypothesis.

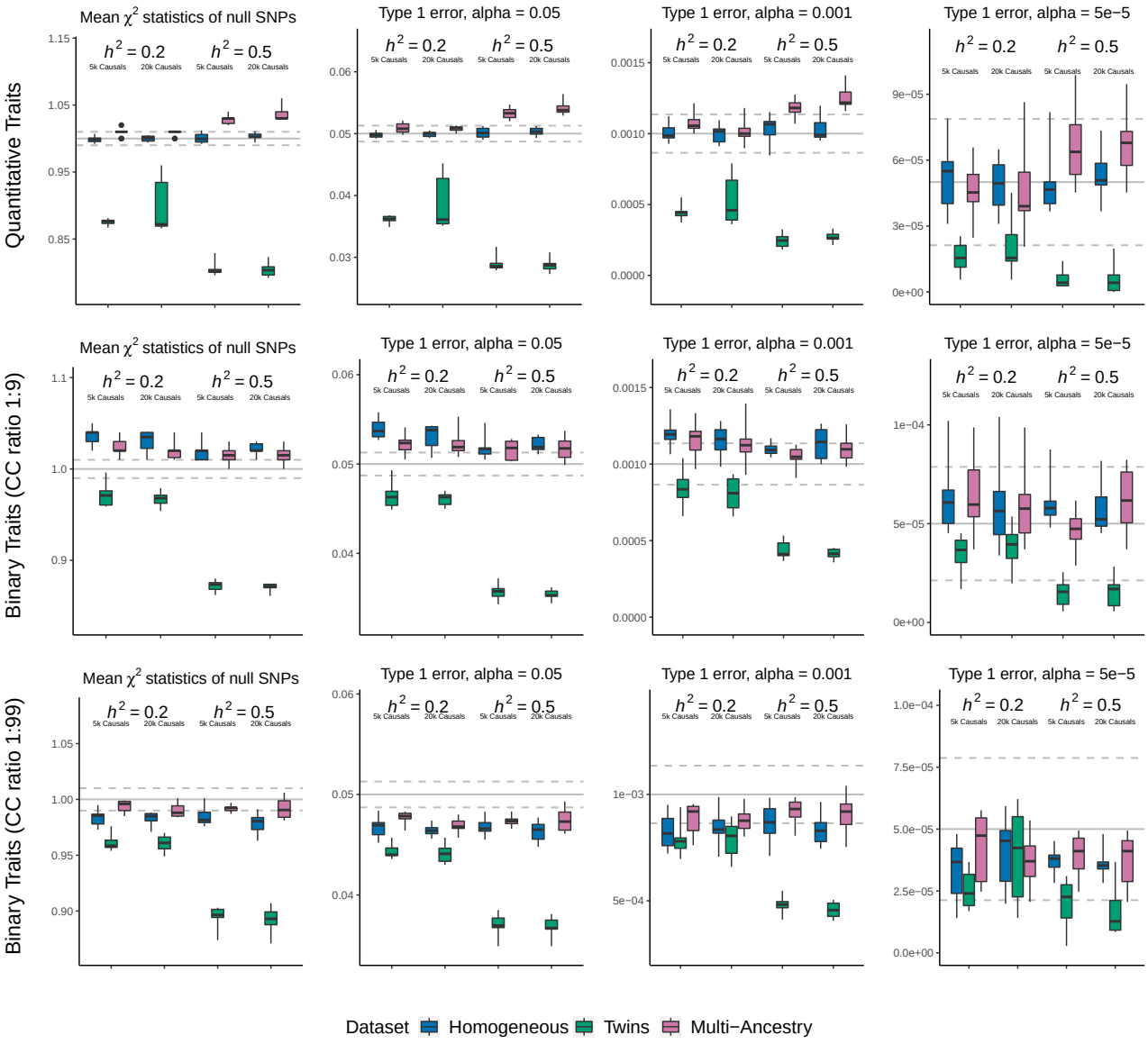
We find that LDAK-KVIK-GBAT has well-controlled type 1 error for all datasets and scenarios considered (although we recognise that its test statistics can be deflated, which is due to the algorithm LDAK-GBAT uses for estimating the null distribution being slightly conservative [22]).




**Supplementary Figure 6: Type 1 error of BOLT-LMM.**

We simulate quantitative phenotypes for the homogeneous, twins and multi-ancestry datasets (which contain 63 k, 63 k and 60 k individuals, respectively). For each dataset, we consider 4 different scenarios, obtained by varying the heritability (0.2 or 0.5) and the number of causal SNPs (5 k or 20 k). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using BOLT-LMM, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null snps (Column 1), and based on the proportions of null SNPs with $p$-value below 0.05, 0.001 or $5 \times 10^{-5}$ (Columns 2, 3 & 4, respectively). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected value of each measure under the null hypothesis, while the dashed grey lines provide a 95% confidence interval (derived by analyzing 200 permuted phenotypes).

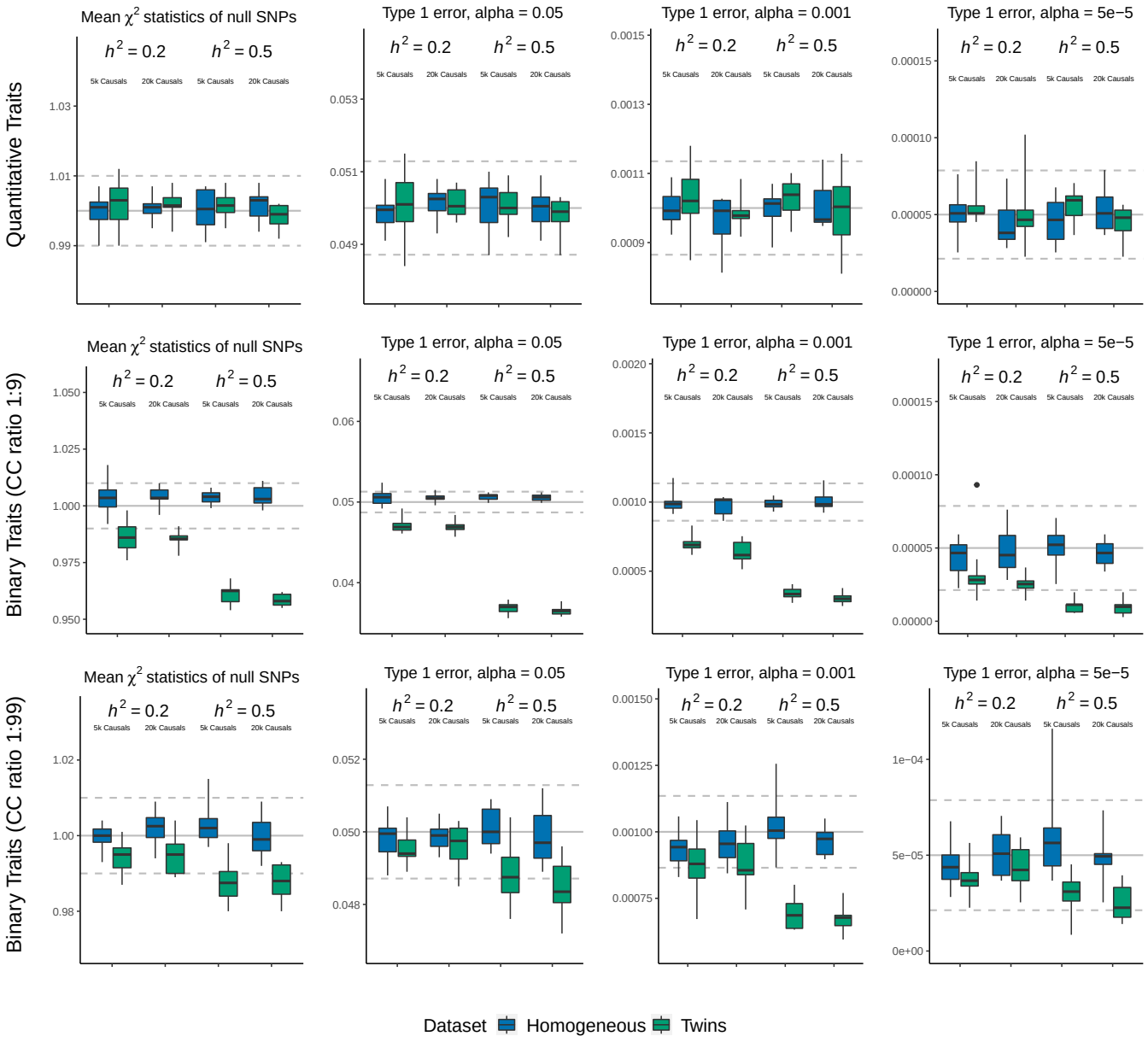
While this figure shows that BOLT-LMM has well-controlled type 1 error for all datasets and scenarios considered, this reflects that we have restricted to phenotypes simulated with $\alpha = -1$ (matching the assumption of BOLT-LMM). In the Main Text, we demonstrate that BOLT-LMM can produce substantially-inflated test statistics when applied to phenotypes simulated with $\alpha = -0.25$, which more closely reflects what we observe for real human traits.




**Supplementary Figure 7: Type 1 error of REGENIE.**

We simulate quantitative and binary phenotypes for the homogeneous, twins and multi-ancestry datasets (which contain 63 k, 63 k and 60 k individuals, respectively). For each dataset, we consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using REGENIE, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null snps (Column 1), and based on the proportions of null SNPs with $p$-value below 0.05, 0.001 or $5 \times 10^{-5}$ (Columns 2, 3 & 4, respectively). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected value of each measure under the null hypothesis, while the dashed grey lines provide a 95% confidence interval (derived by analyzing 200 permuted phenotypes).

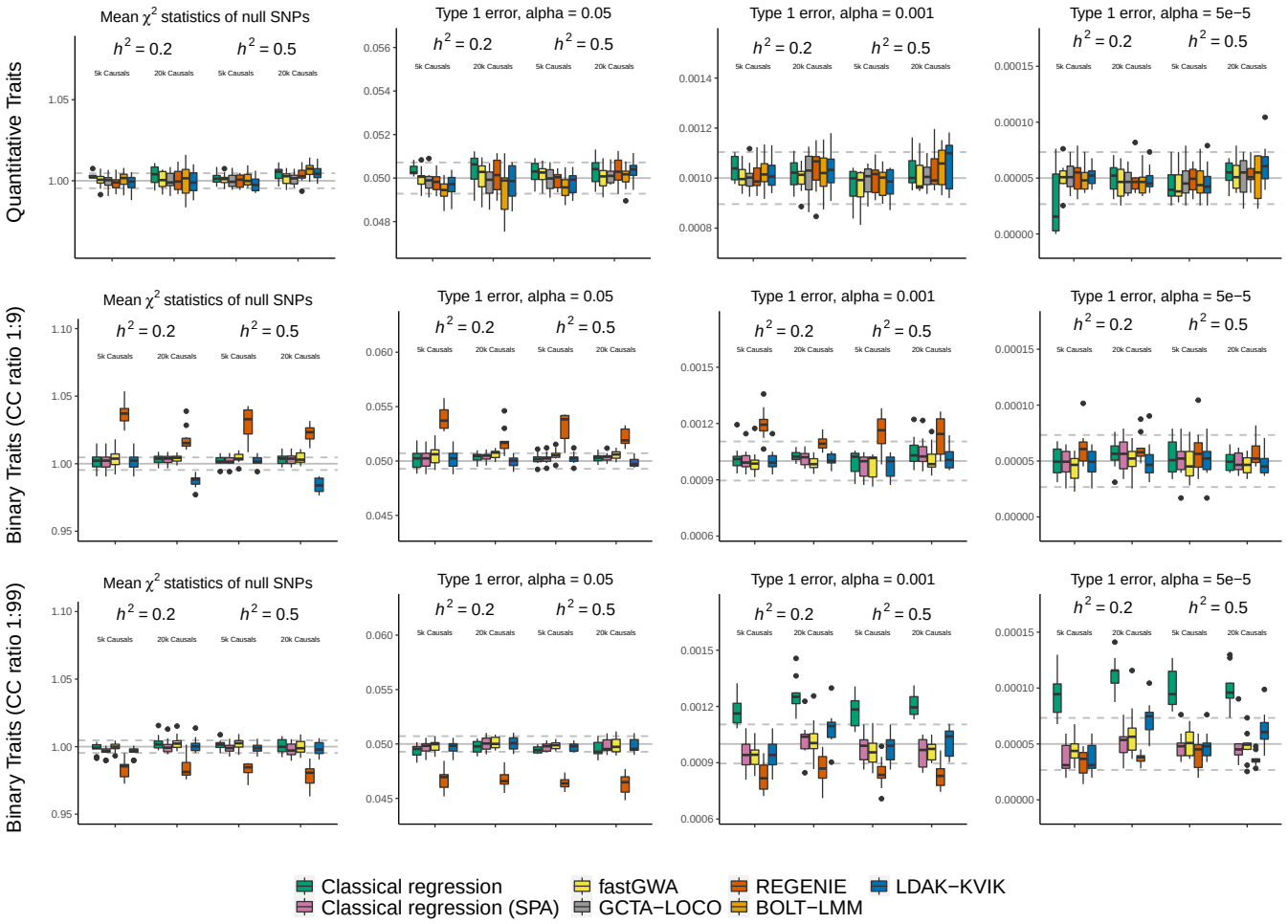
We find that when applied to quantitative phenotypes or rare binary phenotypes, REGENIE has well-controlled type 1 error for all datasets and scenarios considered. However, we find that REGENIE tends to produce inflated test statistics when applied to common binary phenotypes. Furthermore, we find that REGENIE can have deflated test statistics when applied to heterogeneous datasets, which is a consequence of it (implicitly) assuming $\lambda = 1$.




**Supplementary Figure 8: Type 1 error of fastGWA.**

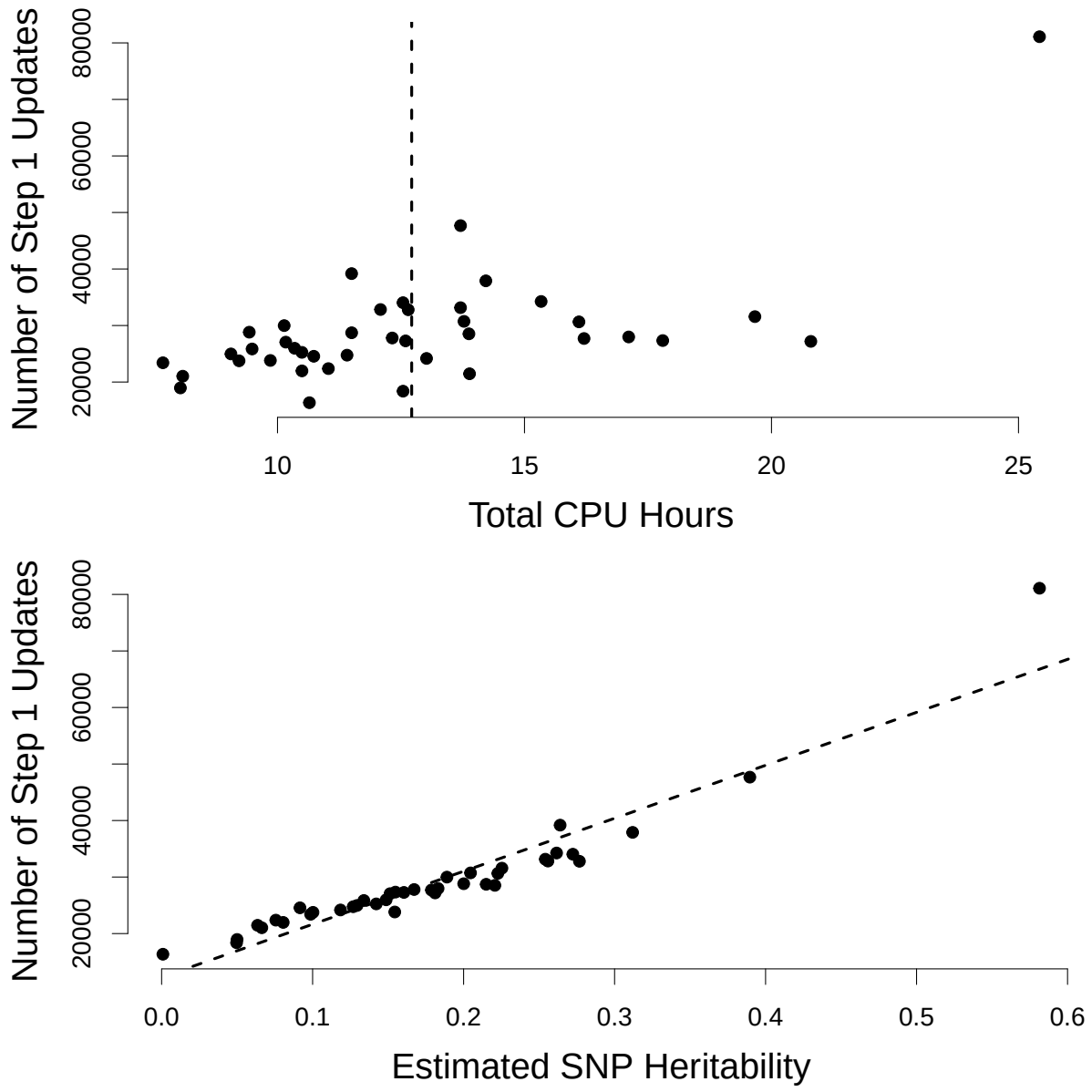
We simulate quantitative and binary phenotypes for the homogeneous and twins datasets (which both contain 63 k, 63 k individuals). For each dataset, we consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using fastGWA, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null snps (Column 1), and based on the proportions of null SNPs with $p$-value below 0.05, 0.001 or $5 \times 10^{-5}$ (Columns 2, 3 & 4, respectively). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected value of each measure under the null hypothesis, while the dashed grey lines provide a 95% confidence interval (derived by analyzing 200 permuted phenotypes).

We find that fastGWA has well-controlled type 1 error for all datasets and scenarios considered (although we note that its test statistics can be deflated when analyzing common binary phenotypes for datasets with high relatedness). [22]).



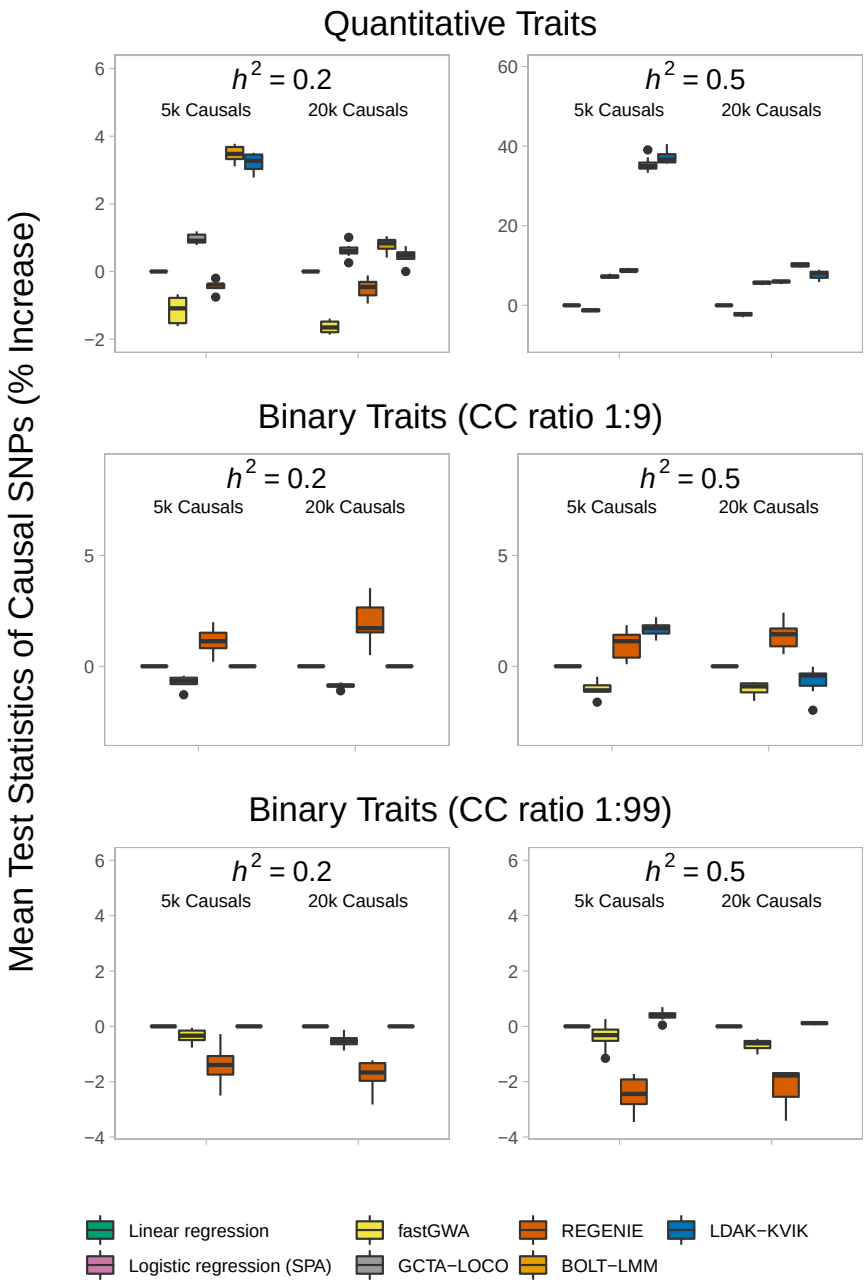

**Supplementary Figure 9: Type 1 error of all MMAA tools for homogeneous data.**
We simulate quantitative and binary phenotypes for the homogeneous dataset (63 k individuals). We consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using classical linear and logistic regression, BOLT-LMM, REGENIE, fastGWA, GCTA-LOCO and LDAK-KVIK, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null snps (Column 1), and based on the proportions of null SNPs with $p$-value below 0.05, 0.001 or $5 \times 10^{-5}$ (Columns 2, 3 & 4, respectively). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected value of each measure under the null hypothesis, while the dashed grey lines provide a 95% confidence interval (derived by analyzing 200 permuted phenotypes).

Although it is difficult to see individuals boxes, this figure nonetheless provides a broad comparison of the type 1 error of different MMAA tools when applied to homogeneous data. We find that the measures of type 1 error are generally close to their expected values under the null hypothesis, with the notable exceptions being the inflation and deflation observed for REGENIE when analyzing common and rare binary phenotypes, respectively (the red boxes in Rows 2 & 3). Furthermore, the final panel demonstrates the importance of using the SPA when analyzing rare binary phenotypes (the green boxes show that basic logistic regression can produce false positives, while the purple boxes show these are avoided by using the SPA).




**Supplementary Figure 10: Runtime of LDAK-KVIK when analyzing 368 k individuals.** We analyze the 40 quantitative UK Biobank phenotypes. The top panel compares the number of variational Bayes updates performed in Step 1 with the total runtime of LDAK-KVIK (i.e., the time taken for both Steps 1 & 2); the dashed vertical line marks the mean runtime across the 40 phenotypes. The bottom panel compares the number of variational Bayes updates performed in Step 1 with the estimated SNP heritability (obtained using our software SumHer [35]); the dashed diagonal line indicates the best fit from regressing number of updates on SNP heritability.

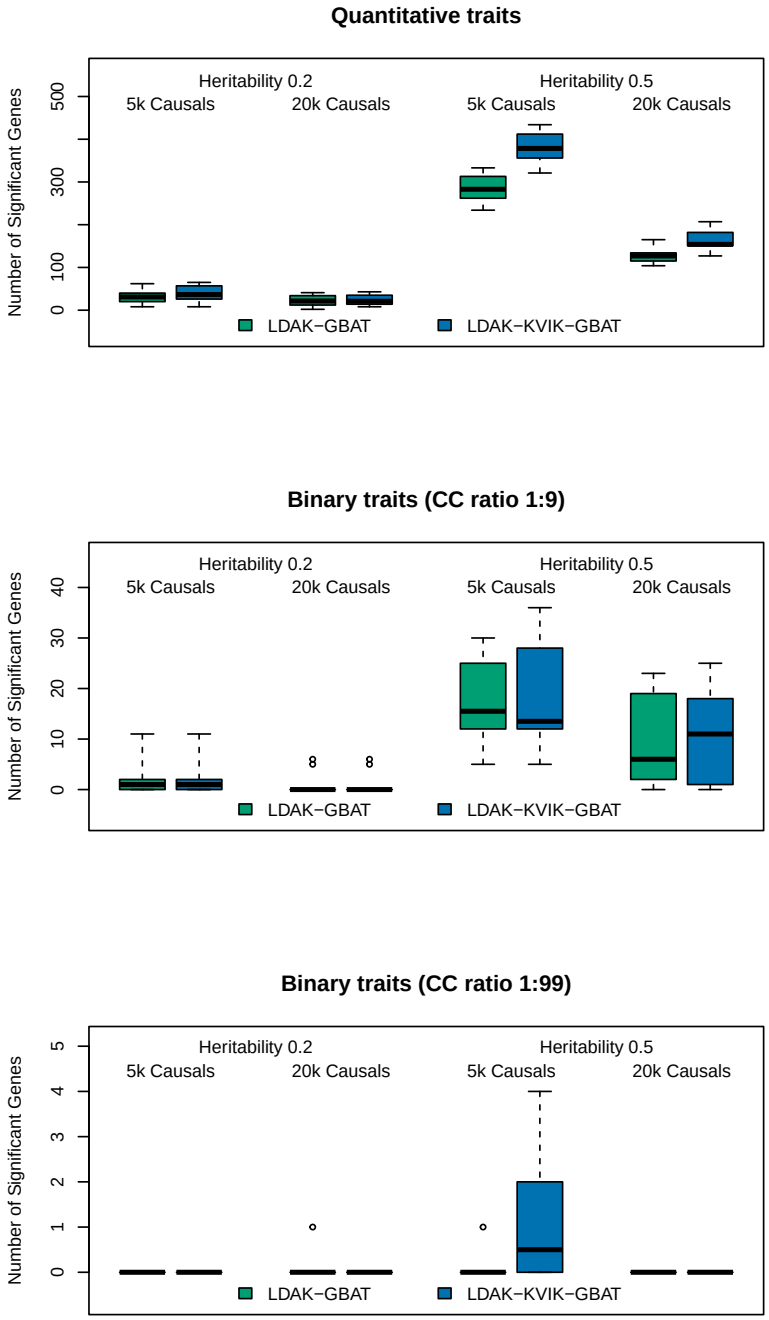
Overall, we see that the total runtime of LDAK-KVIK depends strongly on the number of updates our variational Bayes solver requires to achieve convergence, which in turn is strongly correlated with the SNP heritability of the phenotype. This explains why the slowest runtime was recorded for height (whose heritabilty is about 50% higher than the next most heritable phenotype). Note that the average runtime reported here is lower than for Table 1 in the Main Text, which is mainly because the latter considers only five phenotypes (including height, the phenotype with longest runtime), but also reflects random factors (for the analysis in the main text, we ensured all MMAA tools were run on the same processor, whereas here the choice of processor was left to our high-performance cluster).




**Supplementary Figure 11: Power of MMAA tools for homogeneous data.**

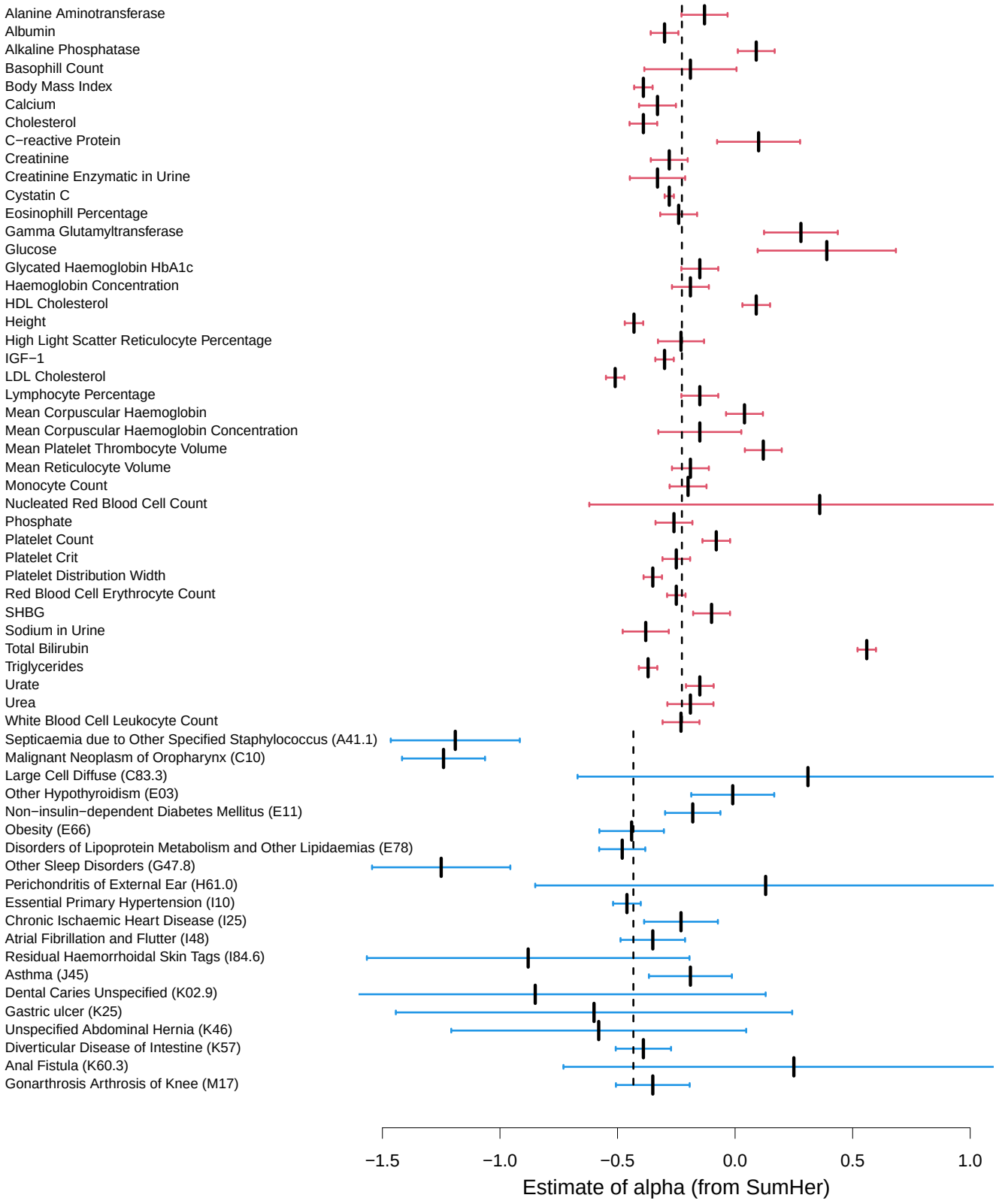
We simulate quantitative and binary phenotypes for the homogeneous dataset (63 k individuals). We consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using fastGWA, GCTA-LOCO, REGENIE, BOLT-LMM and LDAK-KVIK, then measure power based on the mean $\chi^2(1)$ test statistic of causal SNPs, relative to the results from either classical linear regression (quantitative phenotypes) or classical logistic regression (binary phenotypes). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes.

For quantitative phenotypes, we find that BOLT-LMM and LDAK-KVIK tend to have highest power, linear regression and fastGWA have lowest power, while REGENIE and GCTA-LOCO are intermediate. For binary phenotypes, we generally find that all tools have similar power (although REGENIE appears to be more powerful for common binary phenotypes, Supplementary Figure 7 indicates that this is because its test statistics are inflated).



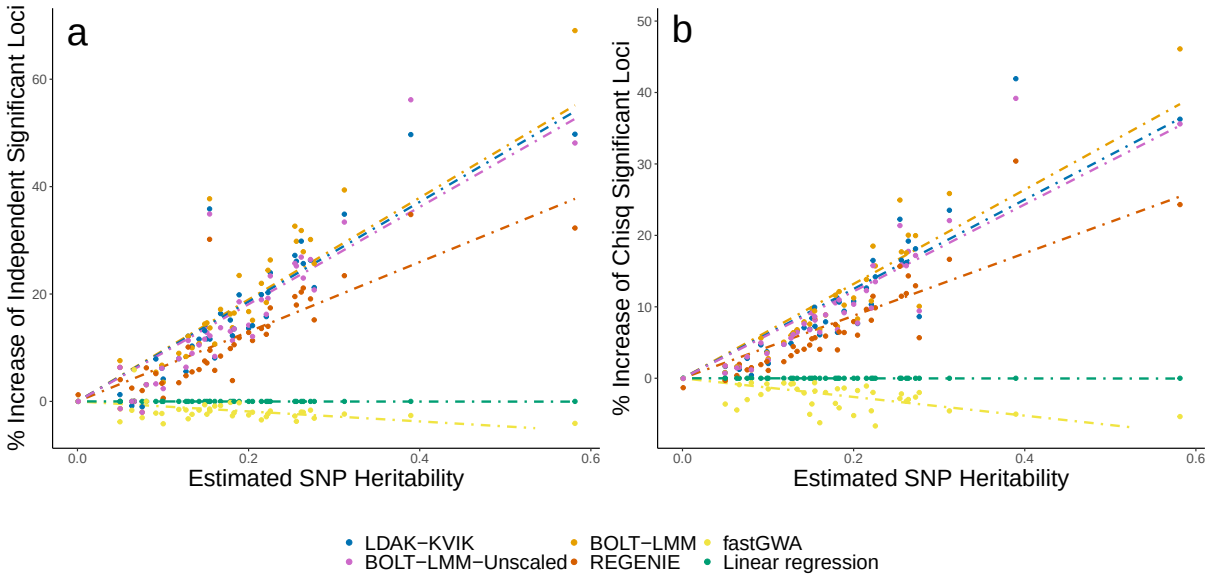

**Supplementary Figure 12: Power of gene-based association testing for homogeneous data.** We simulate quantitative and binary phenotypes for the homogeneous dataset (63 k individuals). We consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform gene-based analysis using either LDAK-GBAT or LDAK-KVIK-GBAT, then measure power based on the number of genes with $p$-value below $0.05/17\,322 = 2.9 \times 10^{-6}$. In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes.

We find that LDAK-KVIK-GBAT tends to be more powerful than LDAK-GBAT for quantitative phenotypes, whereas for binary phenotypes, the two tools have similar power.



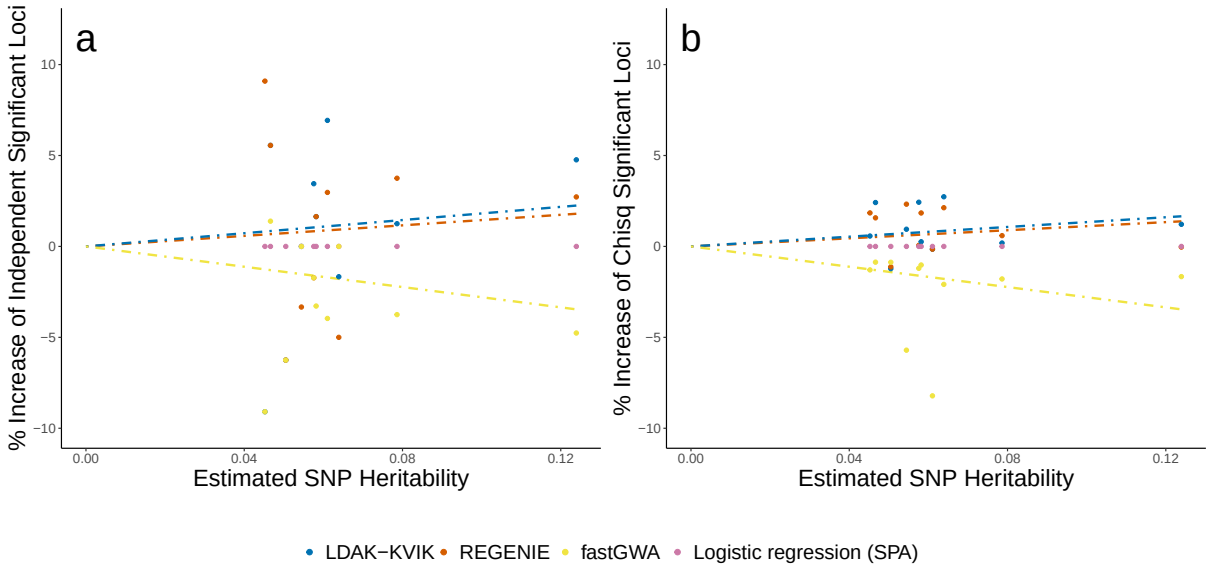

**Supplementary Figure 13: Estimates of $\alpha$ for the UK Biobank phenotypes.**

We report estimates of $\alpha$ for the 40 quantitative, then for the 20 binary phenotypes (obtained using our software SumHer [35]); the vertical dashed lines mark the inverse-variance-weighted mean estimates of $\alpha$ for the two sets of phenotypes.



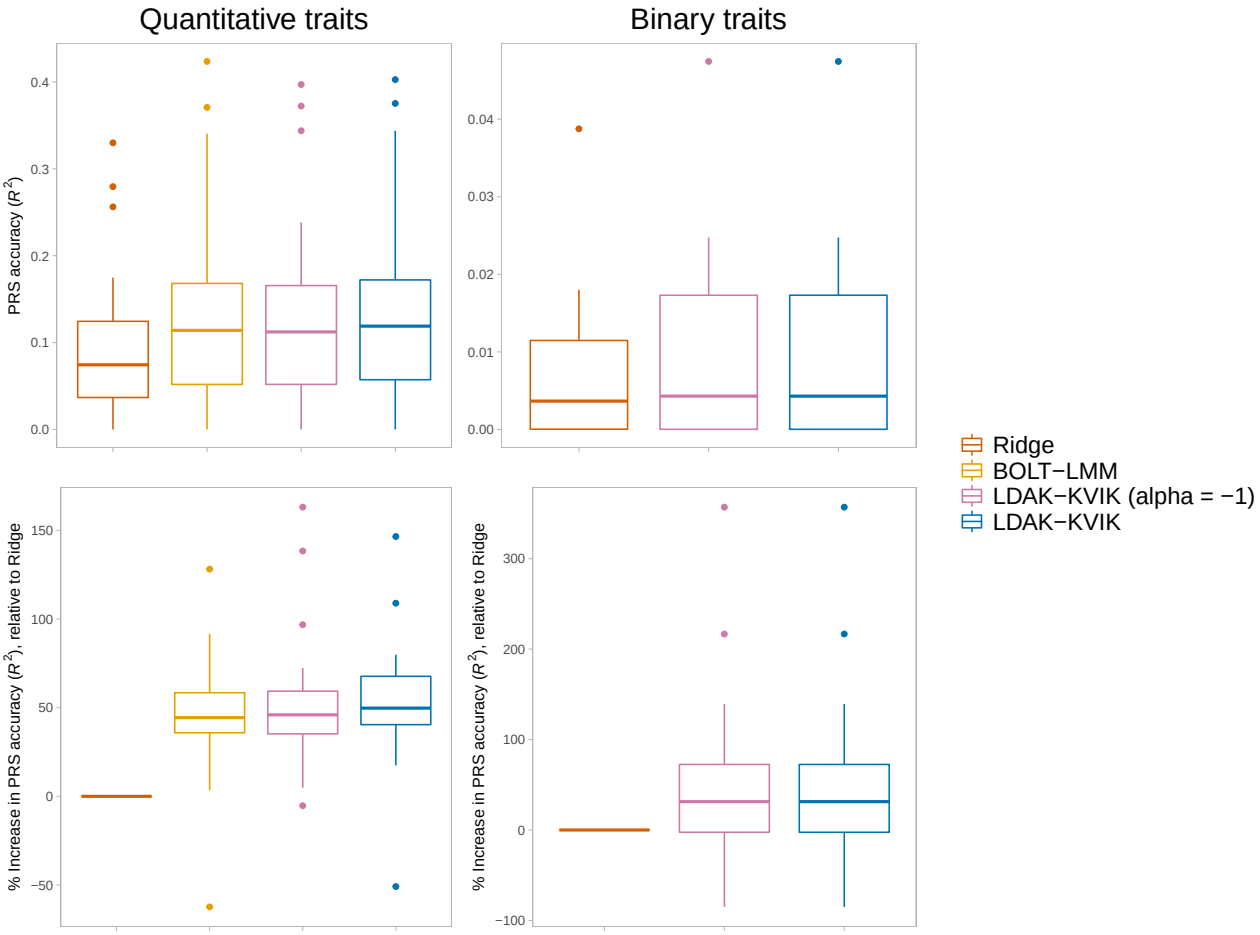

**Supplementary Figure 14: Performance of MMAA tools for 40 quantitative UK Biobank phenotypes.** We analyze each phenotype using BOLT-LMM, BOLT-LMM-Unscaled, REGENIE, fastGWA and LDAK-KVIK, then report (a) the number of independent, genome-wide significant loci (SNPs with $P < 5 \times 10{-}8$, filtered so that no pair within 1 Mb has squared correlation above 0.1), and (b) the mean $\chi^2(1)$ test statistics of SNPs significant from linear regression. For both plots, the values are relative to the results from linear regression. The dashed lines are obtained by regressing the performance of each tool on the estimated SNP heritability (obtained using our software SumHer [35]).

The left panel matches that in the main text, except that we also include the performance of BOLT-LMM. While BOLT-LMM finds most significant loci (slightly ahead of LDAK-KVIK), we believe this reflects that the tool tends to overestimate $\lambda$, and therefore produce inflated test statistics, when applied to phenotypes where the true value of $\alpha$ is substantially above -1 (as shown in Supplementary Figure 13, the mean estimate of $\alpha$ across the 40 phenotypes is -0.23). Therefore, we consider it more fair to compare LDAK-KVIK with BOLT-LMM-Unscaled, where the latter avoids the inflation by setting $\lambda = 1$ (the correct value when analyzing homogeneous data). The right panel shows that the ranking of MMAA tools is robust to changing the way we measure performance.




**Supplementary Figure 15: Performance of MMAA tools for 20 binary UK Biobank phenotypes.** We analyze each phenotype using REGENIE, fastGWA and LDAK-KVIK, then report (a) the number of independent, genome-wide significant loci (SNPs with $P < 5 \times 10{-}8$, filtered so that no pair within 1 Mb has squared correlation above 0.1), and (b) the mean $\chi^2(1)$ test statistics of SNPs significant from linear regression. For both plots, the values are relative to the results from logistic regression. The dashed lines are obtained by regressing the performance of each tool on the estimated SNP heritability (obtained using our software SumHer [35]).

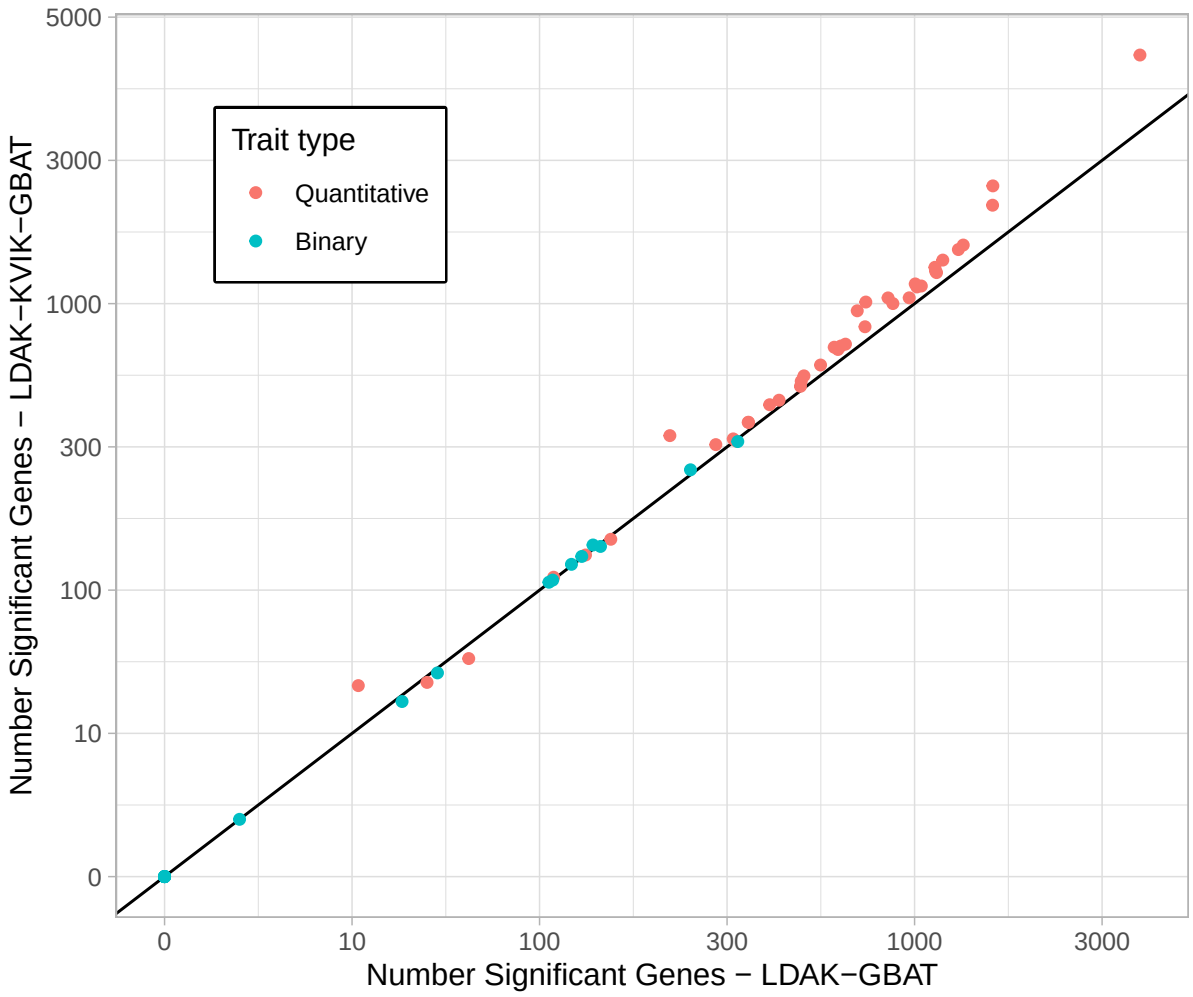
All the lines of best fit are close to the horizontal, which indicates that for the binary phenotypes, the MMAA tools have similar power to classical regression, reflecting that the Step 1 PRS tend to have very low accuracy (Supplementary Figure 16).




**Supplementary Figure 16: Accuracy of Step 1 PRS when analyzing the UK Biobank phenotypes.** We construct PRS for the 40 quantitative phenotypes using a Ridge Regression model, BOLT-LMM, a modified version of LDAK-KVIK that forces $\alpha = -1$, and the default version of LDAK-KVIK. We do the same for the 20 binary phenotypes, except that we exclude BOLT-LMM (which is only designed for quantitative phenotypes). Boxes report the accuracy of each PRS, measured by the squared correlation between predicted and observed phenotypes across 41k samples (distinct from those used to estimate the PRS effect sizes). In each box, the three horizontal lines mark the median and inter-quartile range. Note that the top panels report absolute accuracies, whereas the bottom panels report accuracies relative to those of the Ridge Regression PRS.

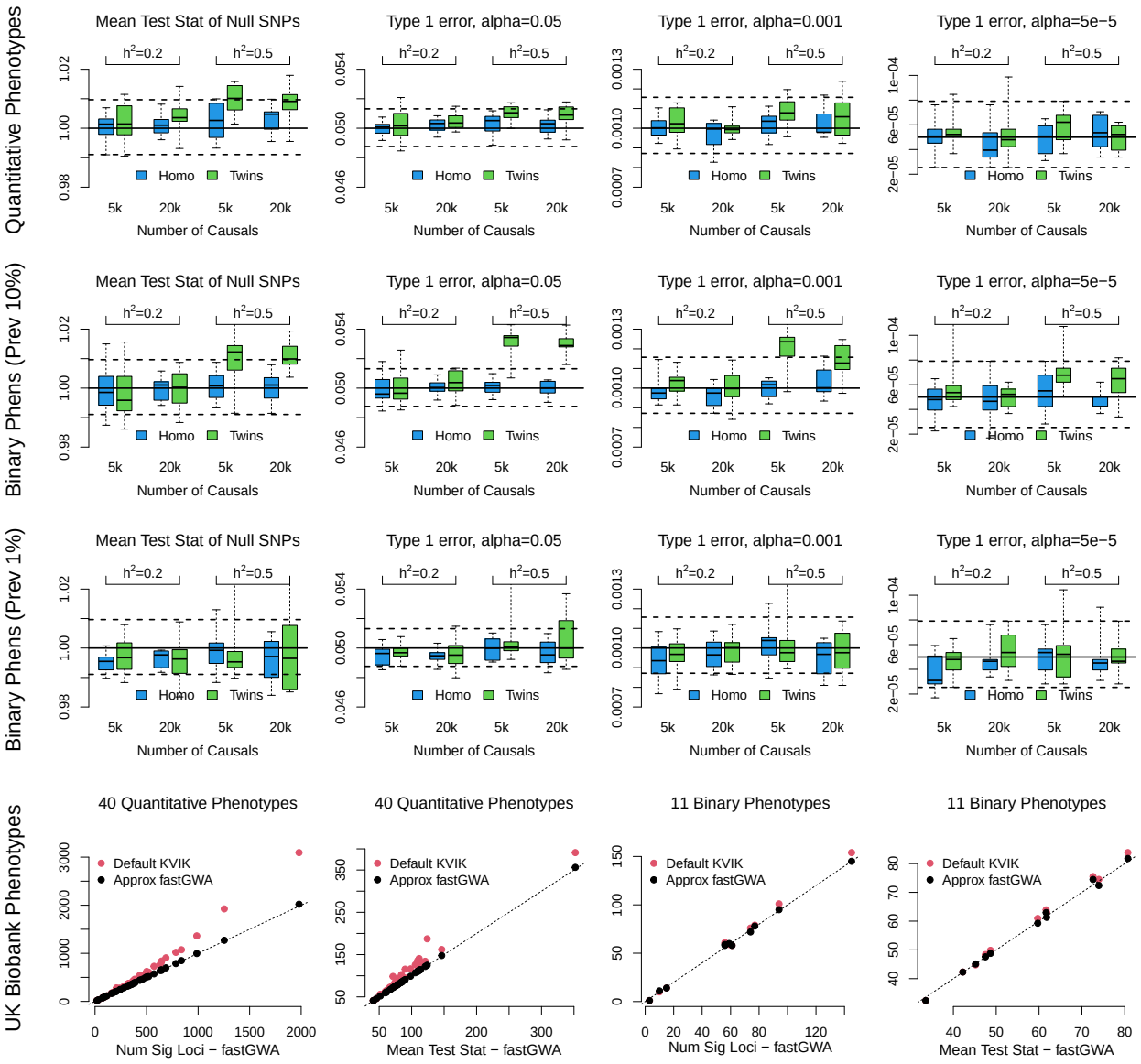
The bottom left panel demonstrates the two main reasons why LDAK-KVIK has slightly increased power relative to BOLT-LMM when analyzing the quantitative phenotypes. Firstly, LDAK-KVIK uses an Elastic Net prior distribution for SNP effect sizes, which results in slightly more accurate Step 1 PRS than BOLT-LMM, which assumes a mixture of normal distributions (this is evident from comparing the second and third boxes). Secondly, LDAK-KVIK estimates $\alpha$ from the data, which improves the accuracy of the Step 1 PRS relative to simply assuming $\alpha = -1$ (evident from comparing the third and fourth boxes). We include the Ridge Regression PRS because these are similar to those constructed by REGENIE (the latter does not report effect sizes, so we can not measure the accuracy of its PRS directly). Therefore, the panel also shows that the substantial power advantage of LDAK-KVIK (and BOLT-LMM) compared to REGENIE, is a consequence of using a mixture prior distribution for SNP effect sizes, instead of an infinitesimal distribution.

The top right panel shows that for the binary phenotypes, both REGENIE and LDAK-KVIK produce Step 1 PRS with low accuracy, explaining why their power is only slightly higher than that of classical logistic regression.




**Supplementary Figure 17: Gene-based association analysis of UK Biobank phenotypes.** Points compare the number of significant genes ($P < 0.05/17\,322 = 2.9 \times 10^{-6}$) from LDAK-GBAT and LDAK-KVIK-GBAT, for each of the 40 quantitative phenotypes (blue) or 20 binary phenotypes (red). The diagonal line marks $y = x$.

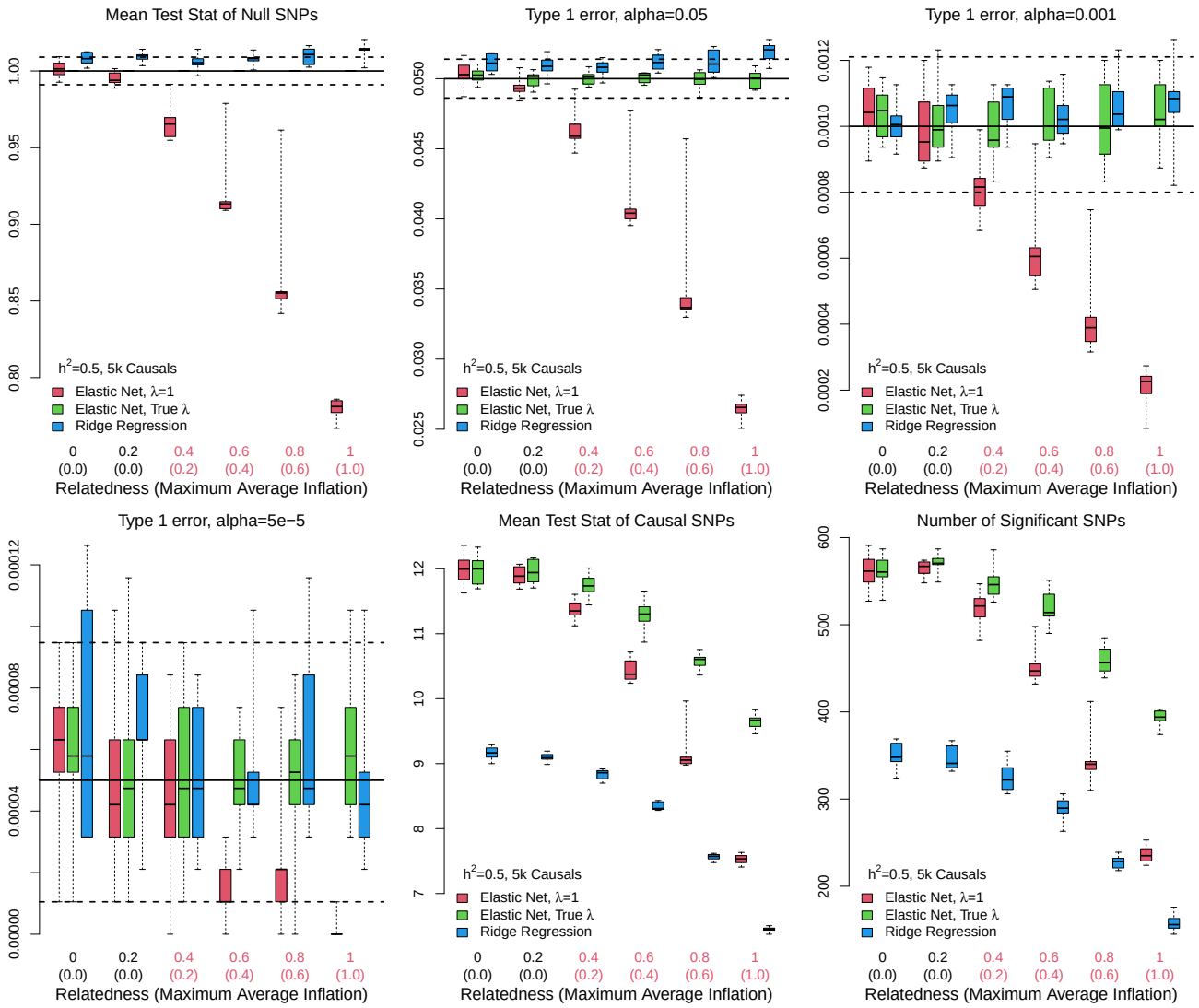
When analyzing the quantitative phenotypes, LDAK-KVIK-GBAT finds on average 18% more significant genes than LDAK-GBAT, whereas for the binary phenotypes, the two tools find similar numbers of significant genes. These result mirror those for single-SNP analysis, where LDAK-KVIK was substantially more powerful than classical regression for quantitative phenotypes, but had similar power for binary phenotypes (reflecting that LDAK-KVIK produces more accurate Step 1 PRS for quantitative phenotypes than for binary phenotypes).




**Supplementary Figure 18: Approximate version of fastGWA.**

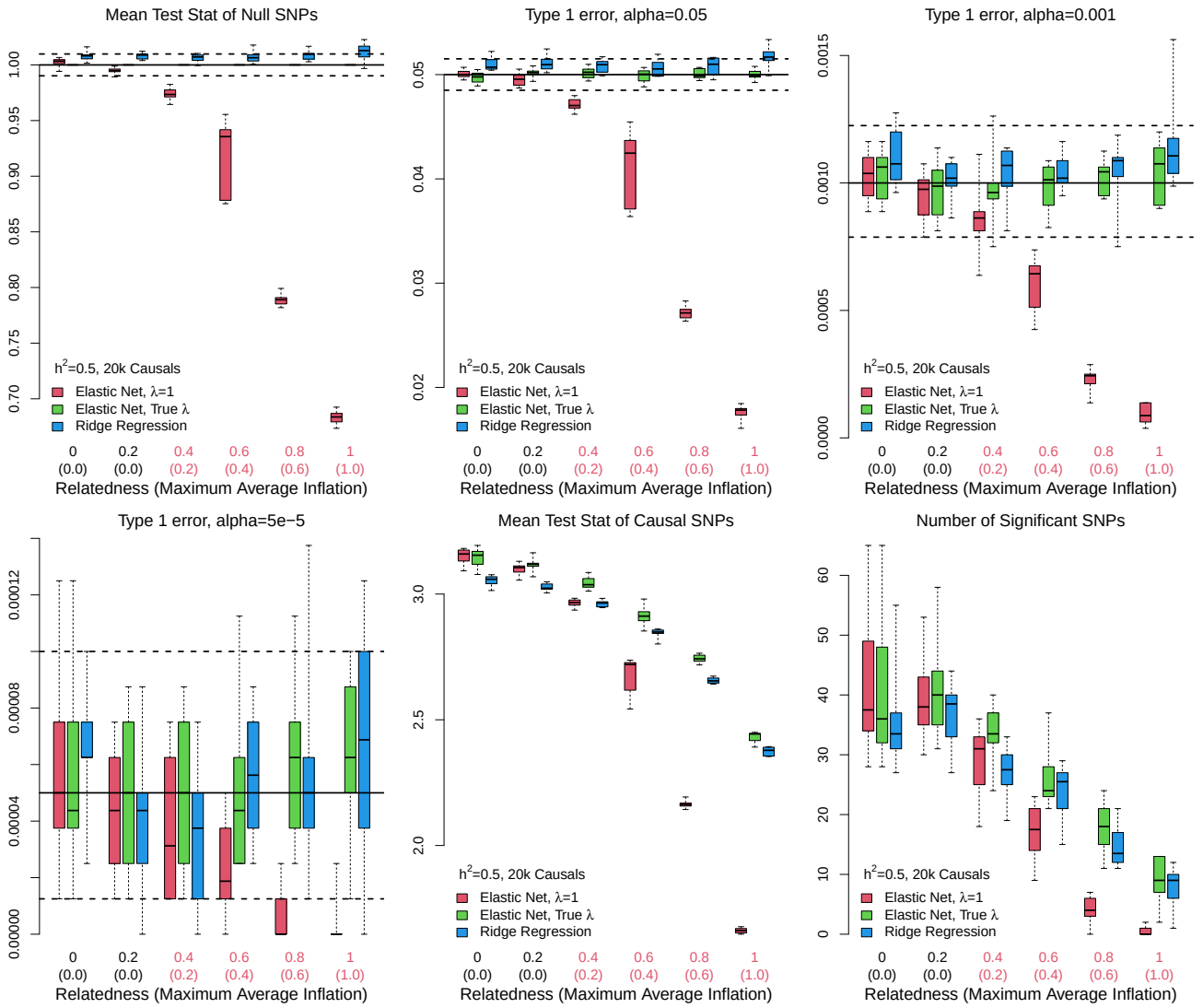
For the top three rows, we simulate quantitative and binary phenotypes for the homogeneous and twins datasets (which both contain 63 k individuals). For each dataset, we consider 12 different scenarios, obtained by varying the heritability (0.2 or 0.5), the number of causal SNPs (5 k or 20 k), and for binary phenotypes, also the prevalence (10% or 1%). When generating causal SNP effect sizes, we assume $\alpha = -1$. We perform single-SNP analysis using the approximate version of fastGWA contained within LDAK-KVIK, then measure the type 1 error based on the mean $\chi^2(1)$ test statistic of null snps (Column 1), and based on the proportions of null SNPs with $p$-value below 0.05, 0.001 or $5 \times 10^{-5}$ (Columns 2, 3 & 4, respectively). In each box, the three horizontal lines mark the median and inter-quartile range across ten phenotypes. The solid grey lines mark the expected value of each measure under the null hypothesis, while the dashed grey lines provide a 95% confidence interval (derived by analyzing 200 permuted phenotypes).

For the bottom row, we apply both the original and approximate versions of fastGWA to the 40 quantitative and 20 binary UK Biobank phenotypes. The black points show that the two versions have very similar performance, both when measured by the number of independent, genome-wide significant loci (SNPs with $P < 5 \times 10{-}8$, filtered so that no pair within 1 Mb has squared correlation above 0.1), and when measured by the mean $\chi^2(1)$ test statistics of SNPs significant from linear regression (for comparison, the red points show the performance of LDAK-KVIK).



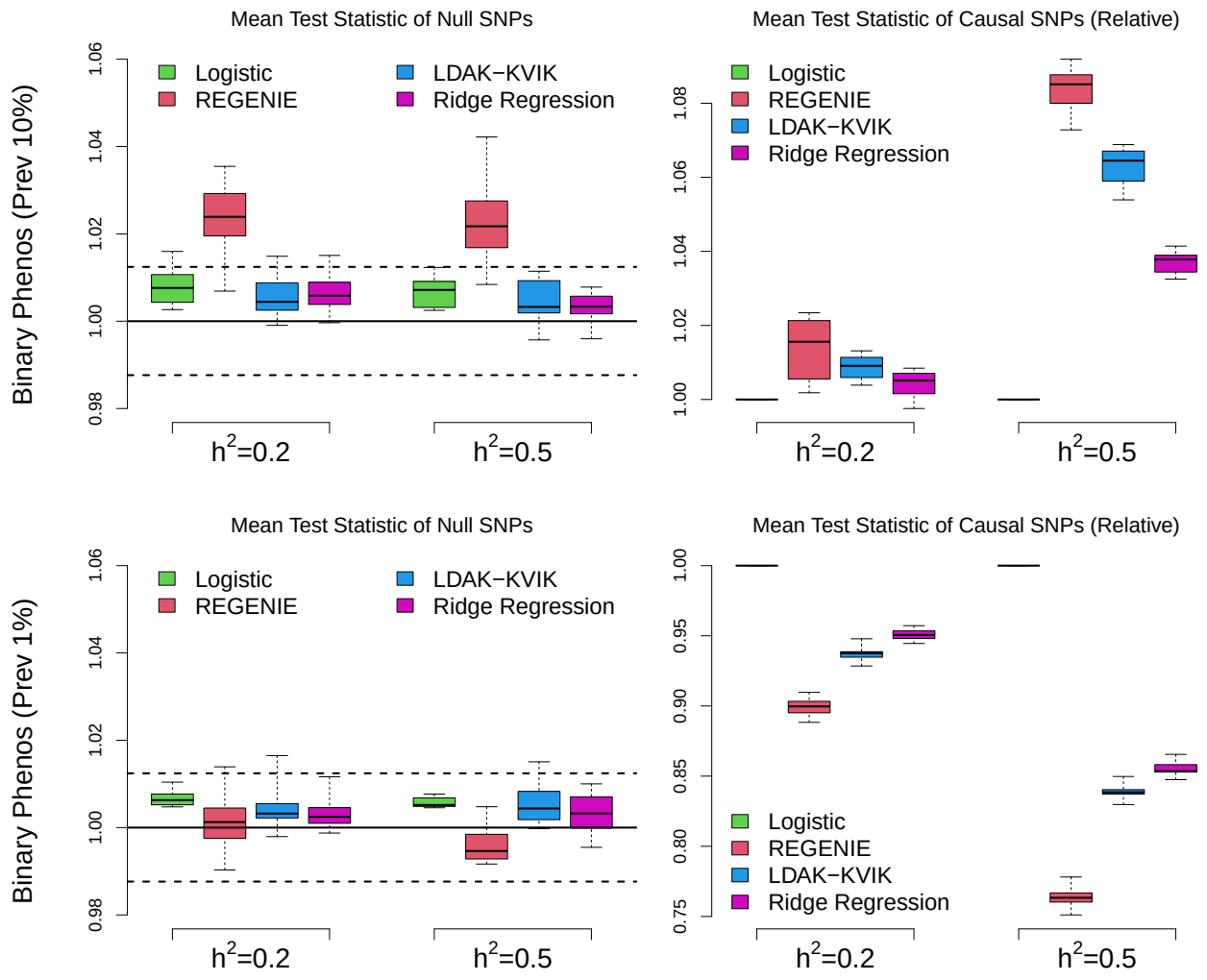

**Supplementary Figure 19: Elastic Net PRS versus Ridge Regression PRS (5 k causal SNPs).** We construct five datasets each containing 70 k individuals and 100 000 SNPs, where the individuals are divided into 35 000 families of size two, and we vary the within-family relatedness from 0 (unrelated) to 1 (pairs of identical twins). Based on the estimated maximum average inflation ($n\bar{\rho}^2$), LDAK-KVIK determines that the two least-related datasets have weak structure ($n\bar{\rho}^2 < 0.1$) and the three most-related datasets have strong structure ($n\bar{\rho}^2 > 0.1$). For each dataset, we simulate phenotypes with heritability 0.5, 5 k causal SNPs and effect sizes generated assuming $\alpha = -1$. Red, green and blue boxes compare, respectively, results from LDAK-KVIK using Elastic Net PRS and forcing $\lambda = 1$, using Elastic Net PRS with true $\lambda = 1$ (i.e., set to the value that ensures perfect control of type 1 error), and using Ridge Regression PRS with $\lambda$ estimated via the Grammar-Gamma Formula. [18]

The red boxes indicate the performance of LDAK-KVIK when applied to datasets with weak structure, whereas the blue boxes indicate its performance when applied to datasets with strong structure. As shown in the final two panels, switching from Elastic Net PRS to Ridge Regression PRS for datasets with high structure results in a lower power to detect causal variants. However, we make this decision because it ensures well-calibrated test statistics (i.e., avoids the deflation observed if we continue to use Elastic Net PRS for datasets with high structure). Further, the reduction in power partially reflects the relatively low number of causal variants (Supplementary Figure 20). Finally, the green boxes show that the ideal solution would be to find a general method for estimating $\lambda$, as then we could use the Elastic Net PRS for all datasets.




**Supplementary Figure 20: Elastic Net PRS versus Ridge Regression PRS (20 k causal SNPs).** We construct five datasets each containing 70 k individuals and 100 000 SNPs, where the individuals are divided into 35 000 families of size two, and we vary the within-family relatedness from 0 (unrelated) to 1 (pairs of identical twins). Based on the estimated maximum average inflation ($n\bar{\rho}^2$), LDAK-KVIK determines that the two least-related datasets have weak structure ($n\bar{\rho}^2 < 0.1$) and the three most-related datasets have strong structure ($n\bar{\rho}^2 > 0.1$). For each dataset, we simulate phenotypes with heritability 0.5, 5 k causal SNPs and effect sizes generated assuming $\alpha = -1$. Red, green and blue boxes compare, respectively, results from LDAK-KVIK using Elastic Net PRS and forcing $\lambda = 1$, using Elastic Net PRS with true $\lambda = 1$ (i.e., set to the value that ensures perfect control of type 1 error), and using Ridge Regression PRS with $\lambda$ estimated via the Grammar-Gamma Formula. [18]

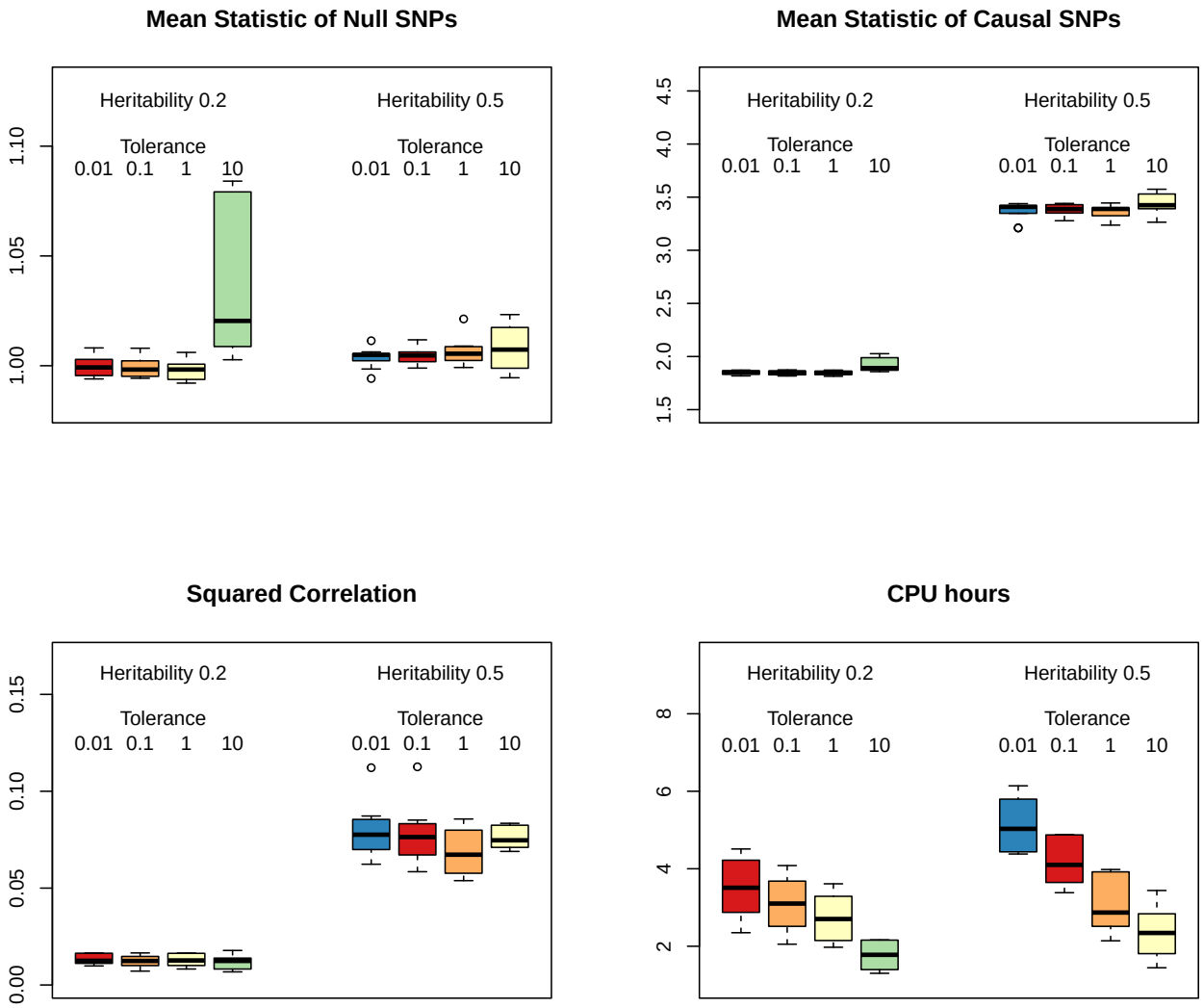
The red boxes indicate the performance of LDAK-KVIK when applied to datasets with weak structure, whereas the blue boxes indicate its performance when applied to datasets with strong structure. As shown in the final two panels, switching from Elastic Net PRS to Ridge Regression PRS for datasets with high structure results in a higher power to detect causal variants. Moreover, this decision ensures well-calibrated test statistics (i.e., avoids the deflation observed if we continue to use Elastic Net PRS for datasets with high structure). However, we recognise that the increase in power partially reflects the relatively high number of causal variants (Supplementary Figure 19). Finally, the green boxes show that the ideal solution would be to find a general method for estimating $\lambda$, as then we could use the Elastic Net PRS for all datasets.




**Supplementary Figure 21: Analyzing ascertained data.**

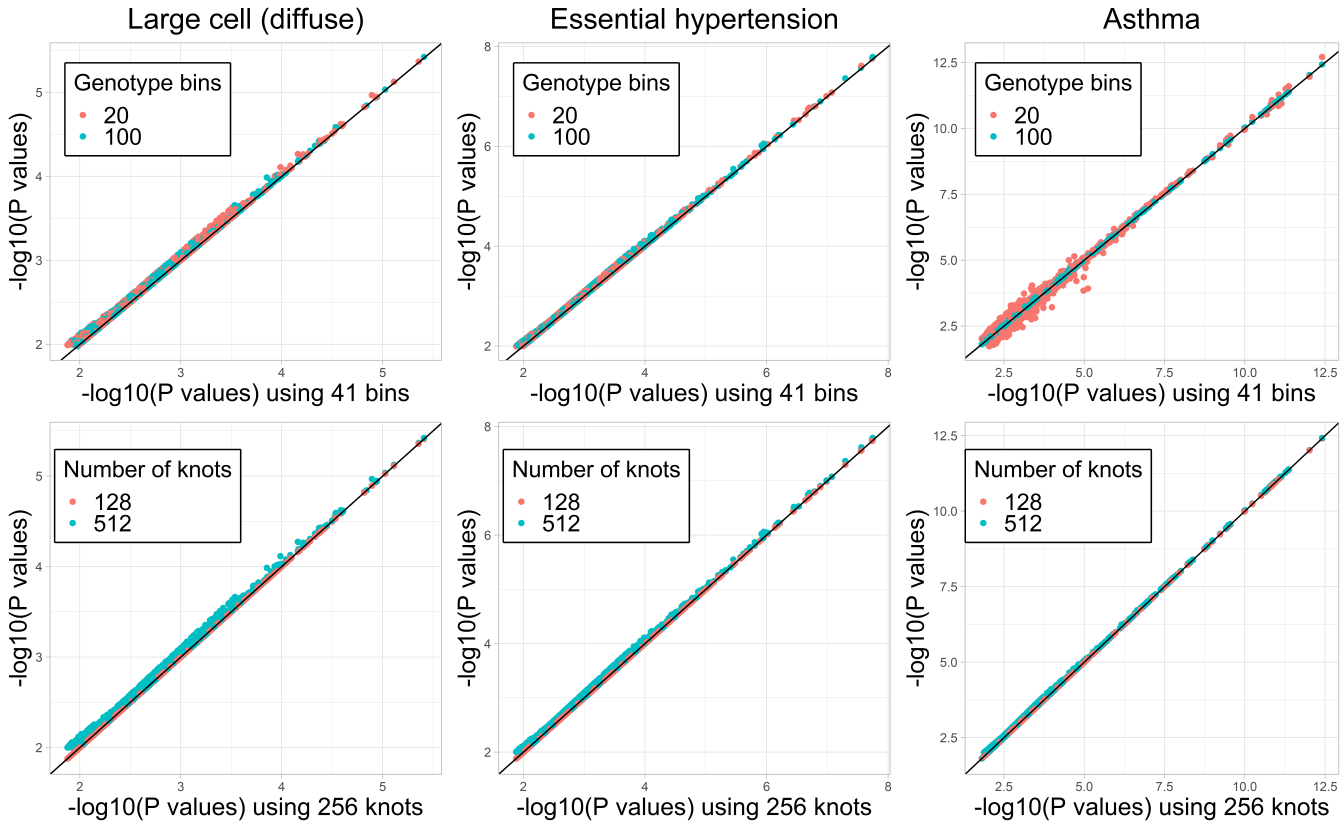
We simulated ascertained datasets as follows. We first generated genotypes for 10 k SNPs and 5.1 M unrelated individuals (all SNPs were in linkage equilibrium). We used this dataset to simulate binary phenotypes with heritability 0.2 or 0.5, 10 k causal SNPs, prevalence 0.1 or 0.01, and effect sizes generated assuming $\alpha = -0.25$. For each phenotype, we then sampled 50 k cases and 50 k controls, and added their genotypes to 50 k randomly-sampled genotypes. The result was multiple datasets, each containing 100 k individuals and 60 k SNPs, where half the individuals were cases and half controls. We analyzed these using classical logistic regression, REGENIE, the default version of LDAK-KVIK and a version of LDAK-KVIK that uses Ridge Regression PRS.

While most tools control type 1 error, we see that REGENIE produces inflated test statistics when applied to common binary phenotypes (this mirrors what we observed for non-ascertained binary phenotypes in Supplementary Figure 7). For this reason, we include the Ridge Regression version of LDAK-KVIK, as this gives an idea how the results from REGENIE would look if the latter was well-calibrated. In terms of power, we see that there is an advantage using MMAA tools for common binary phenotypes, whereas classical logistic regression has highest power for rare binary phenotypes. We believe this may be because the ascertainment causes correlations between the causal SNPs (i.e., long-range linkage disequilibrium), and therefore the LOCO PRS remove some of the causal variation of the target chromosome. Therefore, these results caution against the use of MMAA tools for ascertained rare binary phenotypes (when analyzing approximately homogeneous datasets).




**Supplementary Figure 22: Sensitivity of our variational Bayes solver to the convergence criterion.** By default, our variational Bayes solver ignores chunks that in the previous scan caused the approximate log likelihood to change by less than $n \times 10^{-6}$. Therefore, when analyzing our simulated phenotypes for the homogeneous dataset (63 k individuals), the default tolerance threshold is $63\,000 \times 10^{-6} = 0.063$. Here we reanalyze the quantitative phenotypes varying the tolerance threshold from 0.01 to 10. For each analysis, we measure the mean $\chi^2(1)$ test statistics of null and causal SNPs (top panels), the accuracy of the Step 1 PRS (bottom left panel) and the runtime (bottom right panel). In general, we find that varying the tolerance has limited impact on the accuracy of LDAK-KVIK, albeit lower tolerances lead to longer runtimes. However, the exception is when we set the tolerance to 10 (i.e., approximately 160 times higher than the default), at which point we observe inflated test statistics.

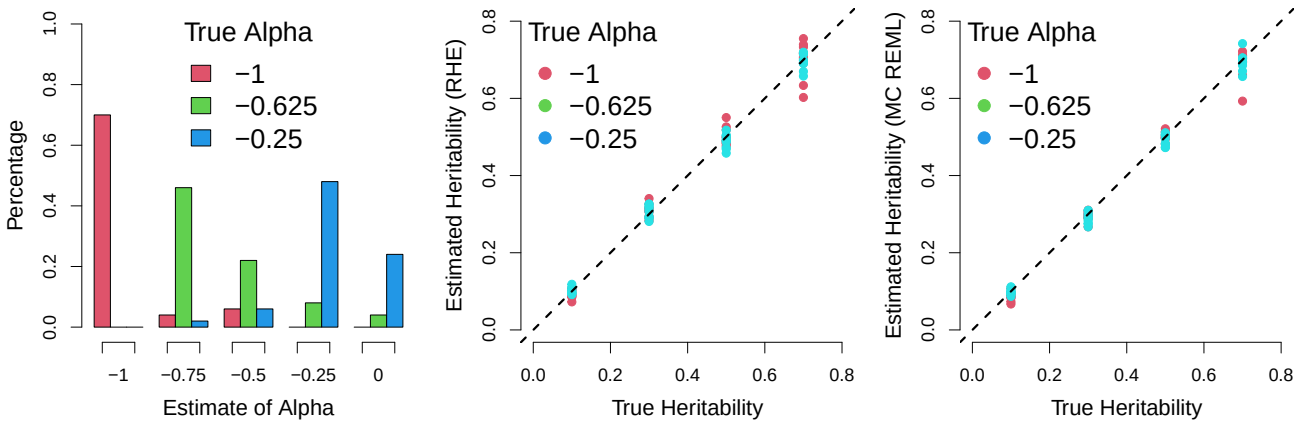
Based on this analysis, we conclude that our default tolerance choice is reasonable.




**Supplementary Figure 23: Sensitivity of our SPA solver to parameter choices.**

Our SPA solver starts by computing realizations of the cumulant generating function and its derivatives for a grid of values, whose size is determined by the numbers of bins and knots. By default, we use 41 bins and 256 knots; this figure shows the impact of changing these numbers to 20 & 100, and to 128 & 512, respectively. Specifically, we analyze three binary UK Biobank phenotypes, using 63 k individuals from the homogeneous dataset, and compare the $-log_{10}$ $p$-values of SNPs for which the SPA was used (those with uncorrected $p$-value below 0.1) when changing the numbers of bins (top panels) or knots (bottom panels). The diagonal line in each panel marks $y = x$. In general, we found that increasing the number of bins or knots had no obvious impact on the resulting $p$-values, whereas reducing either number had a small impact.

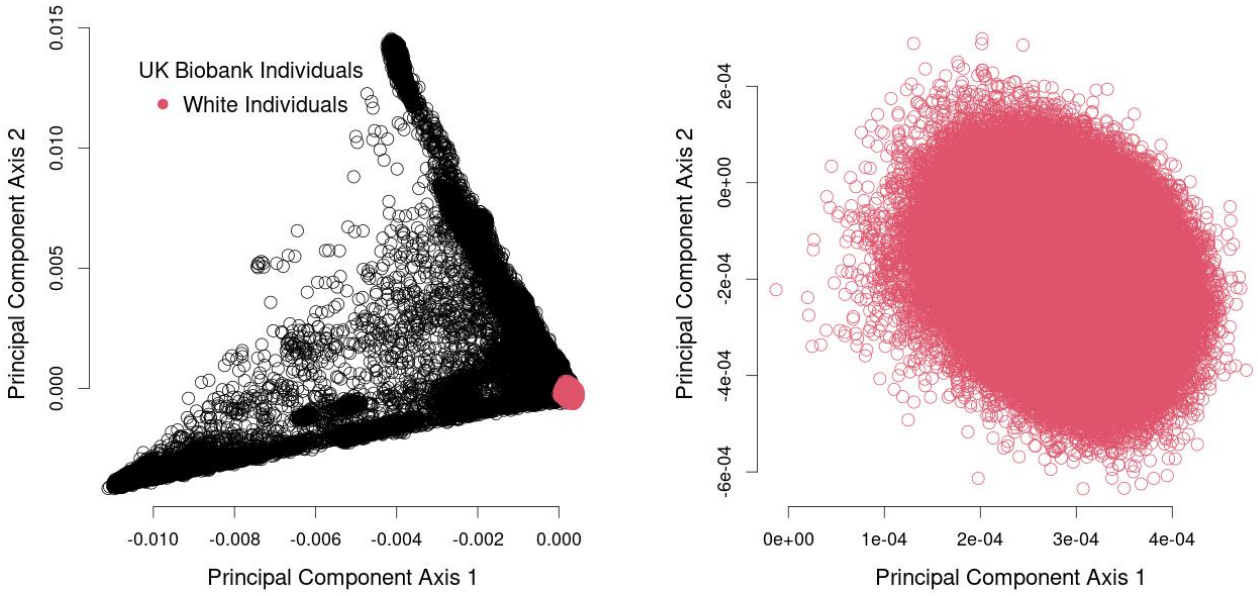
Based on this analysis, we conclude that our default numbers of bins and knots is reasonable.




**Supplementary Figure 24: Randomized Haseman-Elston and Monte Carlo REML produce accurate estimates of $\alpha$ and $h^2$.**

We simulated phenotypes for the homogeneous dataset (63 k individuals) with heritability 0.1, 0.3, 0.5 or 0.7, 10 k causal SNPs and effect sizes generated assuming $\alpha = -1$, $\alpha = -0.625$ or $\alpha = -0.25$. The left and middle panels report, respectively, estimates of $\alpha$ and $h^2$ from randomized Haseman-Elston Regression, while the right panel reports estimates of $h^2$ from Monte Carlo REML (as a reminder, the final estimate of $h^2$ is that from Monte Carlo REML, which is run using the estimate from randomized Haseman-Elston as a starting value).

Based on this analysis, we conclude that randomized Haseman-Elston Regression and Monte Carlo REML are effective at estimating $\alpha$ and $h^2$.




**Supplementary Figure 25: Principal component axes of the UK Biobank data.**

We perform two principal component analyses. For the left panel, we use all 487 k UK Biobank individuals, while for the right panel we use only the 409 k individuals listed in data field 22006 (those who self-identify as being white British, and whom UK Biobank determined "have very similar genetic ancestry based on a principal components analysis of the genotypes"). Note that the individuals used for the second panel are marked in red on the first panel.

| | 690 k SNPs | | | | | | | 10 M SNPs |
|---|---|---|---|---|---|---|---|---|
| | CPU Hours | | | | | Step 1 Operations | | CPU Hours |
| **Phenotype** | **Step 1** | **Step 2** | **Step 2** | **KVIK** | **GBAT** | **Chunks** | **Updates** | **KVIK** |
| Glucose | 10.9 | 0.7 | 2.8 | 11.7 | 14.4 | 10789 | 4.9e+06 | 21.7 |
| Glycated Haemoglobin | 9.4 | 0.6 | 2.8 | 10.0 | 12.8 | 13226 | 6.6e+06 | 18.0 |
| Haemoglobin Conc. | 14.9 | 0.8 | 1.2 | 15.7 | 16.9 | 13485 | 7.4e+06 | 26.4 |
| Height | 26.5 | 1.0 | 2.8 | 27.5 | 30.3 | 21808 | 2.0e+07 | 40.9 |
| HDL Cholesterol | 12.1 | 0.7 | 2.8 | 12.8 | 15.6 | 16967 | 1.0e+07 | 21.7 |
| **Mean** | **14.8** | **0.8** | **2.5** | **15.5** | **18.0** | **15255** | **9.7e+06** | **25.7** |
| **Median** | **12.1** | **0.7** | **2.8** | **12.8** | **15.6** | **13485** | **7.4e+06** | **21.7** |
| J45 Asthma | 17.6 | 3.2 | 2.8 | 20.9 | 23.6 | 11499 | 5.1e+06 | 64.1 |
| I48 Atrial Fibrillation | 20.0 | 4.1 | 2.8 | 24.1 | 26.9 | 10789 | 5.4e+06 | 79.5 |
| I25 Ischaemic Heart Disease | 13.8 | 3.5 | 2.8 | 17.4 | 20.1 | 10792 | 5.4e+06 | 64.9 |
| K02.9 Dental Caries | 13.8 | 2.6 | 2.8 | 16.4 | 19.2 | 10341 | 4.1e+06 | 51.7 |
| I84.6 Skin Tags | 11.3 | 2.5 | 2.8 | 13.8 | 16.6 | 8091 | 3.6e+06 | 47.8 |
| **Mean** | **15.3** | **3.2** | **2.8** | **18.5** | **21.3** | **10302** | **4.7e+06** | **61.6** |
| **Median** | **13.8** | **3.2** | **2.8** | **17.4** | **20.1** | **10789** | **5.1e+06** | **64.1** |

**Supplementary Table 1: LDAK-KVIK runtimes when analyzing 368 k Individuals.**

For each of ten UK Biobank phenotypes (five quantitative and five binary), we report the CPU hours of Steps 1, 2 & 3 of LDAK-KVIK, as well as the total CPU hours for LDAK-KVIK and LDAK-KVIK-GBAT (obtained by summing the times for Steps 1 & 2, and for Steps 1, 2 & 3, respectively). We also report the number of chunks visited and number of updates of $Q_j(\gamma_j)$ by our variational Bayes solver (which determines the total number of operations). All analyses used the white dataset (368 k individuals and 690 k SNPs), except for the final column, which reports the total runtime of LDAK-KVIK when we increase the number of Step 2 SNPs from 690 k to 10 M (i.e., analyze imputed data).

| MMAA Tool | Number of Phenotypes | Quantitative Phenotypes | | | Binary Phenotypes | | |
|---|---|---|---|---|---|---|---|
| | | CPU Hours | Memory (Gb) | Speedup (%) | CPU Hours | Memory (Gb) | Speedup (%) |
| LDAK-KVIK | 1 | 10.4 | 5 | 0 | 18.6 | 4 | 0 |
| | 5 | 20.0 | 6 | 62 | 66.2 | 13 | 29 |
| | 10 | 29.5 | 7 | 72 | 107.6 | 24 | 42 |
| REGENIE | 1 | 30.2 | 14 | 0 | 29.9 | 14 | 0 |
| | 5 | 33.0 | 14 | 78 | 28.7 | 14 | 81 |
| | 10 | 36.4 | 14 | 88 | 31.8 | 14 | 89 |

**Supplementary Table 2: The potential benefit of analyzing multiple phenotypes.**
We considered the UK Biobank phenotypes Glucose and Asthma (the first quantitative and first binary phenotype from Supplementary Table 1). We analyzed each phenotype individually, then analyzed five and ten copies of each phenotype simultaneously (we chose to duplicate phenotypes, instead of selecting unique phenotypes, so that the runtimes focus on the computational benefits of analyzing multiple phenotypes, rather than the composition of phenotypes). All analyses were performed using the white dataset (368 k individuals and 690 k SNPs). For each analysis, we report runtime, memory usage and per-phenotype speedup (the latter is calculated with respect to the single-phenotype analysis).

We see that when applying LDAK-KVIK to quantitative phenotypes, it is possible to obtain a substantial reduction in total runtime by analyzing multiple phenotypes. For example, it took LDAK-KVIK 29.5 CPU hours to analyze ten copies of Glycated Haemoglobin, which is 72% less than the time it would take to individually analyze the trait ten times ($10 \times 10.4 = 104$ CPU hours). For binary phenotypes, the reduction is more modest, reflecting that LDAK-KVIK requires a separate weight matrix $D$ for each phenotype (this prevents some algebraic operations being applied to multiple phenotypes at once). In general, we found that larger speedups are possible with REGENIE, both for quantitative and binary phenotypes. However, it is worth noting that for these analyses, it was necessary to use the low memory feature (activated by adding the option `--lowmem`), as otherwise the memory would increase linearly (e.g., to analyze ten phenotypes would require over 100 Gb). Using this feature caused the baseline runtime to increase substantially (e.g., REGENIE took 30 CPU hours to analyze a single quantitative phenotype, whereas without the low memory option it took only 11 CPU hours), thus reducing the impact of the speedup.

| MMAA Tool | Analyzing 368 k Individuals and 690 k SNPs | | | |
|---|---|---|---|---|
| | **Regression Model** | **Correction** | **CPU Hours** | **Difference** |
| LDAK-KVIK | Linear | None | 0.8 | NA |
| | Logistic | None | 0.8 | 0 |
| | Logistic | Our novel SPA solver | 1.3 | 0.5 |
| | Logistic | A naïve SPA solver | 1.8 | 1.0 |
| REGENIE | Linear | None | 2.3 | NA |
| | Logistic | None | 2.0 | 0 |
| | Logistic | fastSPA | 5.0 | 3.0 |
| | Logistic | Approximate Firth Regression | 5.4 | 3.4 |

**Supplementary Table 3: Performance of our novel SPA solver.**

We analyzed a binary phenotype for the white dataset (368 k individuals and 690 k SNPs). We first used LDAK-KVIK to perform linear regression (without a SPA solver), followed by three versions of logistic regression: without a SPA solver, using our novel SPA solver, and using a naïve SPA solver (we implemented the non-fast version of SPA from the fastSPA publication [30]). We next used REGENIE to perform linear regression (without a SPA solver), followed by three versions of logistic regression: without a SPA solver, using their implementation of fastSPA, and using an approximate version of Firth Regression (the recommendation of the authors of REGENIE [6]). As well as reporting total runtime, for the analyses that use logistic regression, we also report the difference in runtime relative to non-corrected logistic regression (which is an estimate of the extra time required to compute corrected $p$-values).

We found it challenging to compare our novel SPA solver with existing solvers. This is because the runtime of each analysis depends not only on the time to perform the SPA, but also on factors such as the time to perform regular logistic regression, and the time to read in and manipulate the data. For example, while the above values suggest that our SPA solver is much faster than the implementation of fastSPA within REGENIE (1.3 CPU hours versus 5.0 CPU hours), we note that LDAK-KVIK performs non-corrected logistic regression over twice as fast as REGENIE (0.8 CPU hours versus 2.0 hours).

Nonetheless, we believe that these results show that our novel SPA solver has above-average performance. When considering absolute runtimes, we see that LDAK-KVIK can perform SPA-corrected logistic regression for a large dataset within a few CPU hours. Meanwhile, when considering relative runtimes, our solver is about twice as fast as a naïve SPA solver, and faster than the approximate Firth Regression solver used by REGENIE. Furthermore, we note that aside from its reduced runtime, our solver has the advantages that it is empirical, meaning that it can be applied to any type of phenotype, and that it does not require sparse genotypes, meaning that, for example, it can be applied to dosage data.

| **Ethnicity** | White Dataset | Homogeneous Dataset | Twins Dataset | Multi-Ancestry Dataset |
|---|---|---|---|---|
| White | 0 | 0 | 0 | 490 |
| British | 367981 | 63000 | 31500 (duplicated) | 8751 |
| Irish | 0 | 0 | 0 | 11438 |
| Any other White background | 0 | 0 | 0 | 14209 |
| Mixed | 0 | 0 | 0 | 44 |
| White and Black Caribbean | 0 | 0 | 0 | 526 |
| White and Black African | 0 | 0 | 0 | 355 |
| White and Asian | 0 | 0 | 0 | 722 |
| Any other Mixed background | 0 | 0 | 0 | 881 |
| Asian or Asian British | 0 | 0 | 0 | 38 |
| Indian | 0 | 0 | 0 | 5070 |
| Pakistani | 0 | 0 | 0 | 1572 |
| Bangladeshi | 0 | 0 | 0 | 198 |
| Any other Asian background | 0 | 0 | 0 | 1570 |
| Black or Black British | 0 | 0 | 0 | 25 |
| Caribbean | 0 | 0 | 0 | 3834 |
| African | 0 | 0 | 0 | 2870 |
| Any other Black background | 0 | 0 | 0 | 108 |
| Chinese | 0 | 0 | 0 | 1338 |
| Other ethnic group | 0 | 0 | 0 | 3891 |
| Prefer not to answer | 0 | 0 | 0 | 1428 |
| Do not know | 0 | 0 | 0 | 188 |
| Not reported | 0 | 0 | 0 | 473 |
| **Total** | 367981 | 63000 | 63000 | 60019 |

**Supplementary Table 4: Ethnic compositions of the UK Biobank datasets.**

This table provides a breakdown of our four UK Biobank datasets, based on the ethnic labels provided in data field 22006.

| Phenotype | Data Field | Sample Size | Mean | SD | Test Samples |
|---|---|---|---|---|---|
| Alanine Aminotransferase | 30620 | 350684 | 23.53 | 14.10 | 39032 |
| Albumin | 30600 | 321188 | 45.23 | 2.61 | 35786 |
| Alkaline Phosphatase | 30610 | 350811 | 83.76 | 26.52 | 39057 |
| Basophill Count | 30160 | 356431 | 0.03 | 0.05 | 39591 |
| Body Mass Index | 21001 | 366801 | 27.41 | 4.75 | 40772 |
| Calcium | 30680 | 321072 | 2.38 | 0.09 | 35770 |
| Cholesterol | 30690 | 350793 | 5.71 | 1.15 | 39054 |
| C-reactive Protein | 30710 | 350036 | 2.60 | 4.40 | 38972 |
| Creatinine | 30700 | 350616 | 72.34 | 17.93 | 39036 |
| Creatinine Enzymatic in Urine | 30510 | 357452 | 8800.26 | 5747.65 | 39738 |
| Cystatin C | 30720 | 350758 | 0.91 | 0.17 | 39058 |
| Eosinophill Percentage | 30210 | 356436 | 2.56 | 1.85 | 39591 |
| Gamma Glutamyltransferase | 30730 | 350600 | 37.39 | 41.81 | 39049 |
| Glucose | 30740 | 320821 | 5.12 | 1.21 | 35750 |
| Glycated Haemoglobin HbA1c | 30750 | 350802 | 35.97 | 6.52 | 38941 |
| Haemoglobin Concentration | 30020 | 357065 | 14.20 | 1.23 | 39654 |
| HDL Cholesterol | 30760 | 321043 | 1.45 | 0.38 | 35772 |
| Height | 50 | 367194 | 168.72 | 9.25 | 40808 |
| High Light Scatter Reticulocyte Percentage | 30290 | 351312 | 0.40 | 0.33 | 39061 |
| IGF-1 | 30770 | 348904 | 21.38 | 5.66 | 38833 |
| LDL Cholesterol | 30780 | 350138 | 3.57 | 0.87 | 38992 |
| Lymphocyte Percentage | 30180 | 356436 | 28.63 | 7.36 | 39591 |
| Mean Corpuscular Haemoglobin | 30050 | 357062 | 31.55 | 1.82 | 39654 |
| Mean Corpuscular Haemoglobin Concentration | 30060 | 357059 | 34.54 | 1.06 | 39653 |
| Mean Platelet Thrombocyte Volume | 30100 | 357056 | 9.32 | 1.08 | 39655 |
| Mean Reticulocyte Volume | 30260 | 351312 | 105.81 | 7.77 | 39061 |
| Monocyte Count | 30130 | 356431 | 0.48 | 0.22 | 39591 |
| Nucleated Red Blood Cell Count | 30170 | 356423 | 0.00 | 0.03 | 39590 |
| Phosphate | 30810 | 320580 | 1.16 | 0.16 | 35719 |
| Platelet Count | 30080 | 357061 | 253.41 | 59.89 | 39655 |
| Platelet Crit | 30090 | 357057 | 0.23 | 0.05 | 39655 |
| Platelet Distribution Width | 30110 | 357056 | 16.49 | 0.52 | 39655 |
| Red Blood Cell Erythrocyte Count | 30010 | 357064 | 4.51 | 0.41 | 39655 |
| SHBG | 30830 | 318117 | 51.86 | 27.61 | 35437 |
| Sodium in Urine | 30530 | 356709 | 76.22 | 43.61 | 39640 |
| Total Bilirubin | 30840 | 349347 | 9.13 | 4.42 | 38879 |
| Triglycerides | 30870 | 350514 | 1.76 | 1.02 | 39025 |
| Urate | 30880 | 350359 | 309.28 | 80.28 | 39015 |
| Urea | 30670 | 350556 | 5.44 | 1.39 | 39025 |
| White Blood Cell Leukocyte Count | 30000 | 357060 | 6.89 | 2.07 | 39655 |

**Supplementary Table 5: Baseline characteristics of the 40 quantitative UK Biobank phenotypes.** The sample sizes, means and standard deviations correspond to phenotyped individuals within the white dataset (367 981 white British individuals), which we used for most of our analyses of UK Biobank phenotypes. The test samples are the number of phenotyped individuals within the independent test dataset (40 887 white British individuals), which we used for measuring the accuracy of Step 1 PRS.

| Phenotype | ICD-10 Code | Prevalence |
|---|---|---|
| Essential Primary Hypertension | I10 | 0.2871 |
| Disorders of Lipoprotein Metabolism and Other Lipidaemias | E78 | 0.1422 |
| Diverticular Disease of Intestine | K57 | 0.1227 |
| Chronic Ischaemic Heart Disease | I25 | 0.0941 |
| Asthma | J45 | 0.0899 |
| Gonarthrosis Arthrosis of Knee | M17 | 0.0773 |
| Non-insulin-dependent Diabetes Mellitus | E11 | 0.0715 |
| Atrial Fibrillation and Flutter | I48 | 0.0699 |
| Obesity | E66 | 0.0640 |
| Other Hypothyroidism | E03 | 0.0557 |
| Gastric ulcer | K25 | 0.0163 |
| Dental Caries Unspecified | K02.9 | 0.0102 |
| Residual Haemorrhoidal Skin Tags | I84.6 | 0.0098 |
| Anal Fistula | K60.3 | 0.0036 |
| Large Cell Diffuse | C83.3 | 0.0025 |
| Unspecified Abdominal Hernia | K46 | 0.0018 |
| Perichondritis of External Ear | H61.0 | 0.0012 |
| Septicaemia due to Other Specified Staphylococcus | A41.1 | 0.0008 |
| Other Sleep Disorders | G47.8 | 0.0004 |
| Malignant Neoplasm of Oropharynx | C10 | 0.0002 |

**Supplementary Table 6: Baseline characteristics of the 20 binary UK Biobank phenotypes.** We obtained ICD-10 codes from data field 41270. Note that there are no missing values, because everyone not recorded as being a case for a particular phenotype (i.e., not having the corresponding ICD-10 code) is automatically assumed to be a control.

# References


[1] Loh, P.-R. *et al.* Efficient Bayesian mixed-model analysis increases association power in large cohorts. *Nature genetics* **47**, 284–290 (2015).

[2] Speed, D., Holmes, J. & Balding, D. J. Evaluating and improving heritability models using summary statistics. *Nature Genetics* **52**, 458–462 (2020).

[3] Yang, J. *et al.* Common SNPs explain a large proportion of the heritability for human height. *Nat. Genet.* **42**, 565–569 (2010).

[4] Speed, D., Hemani, G., Johnson, M. & Balding, D. Improved heritability estimation from genome-wide SNP data. *Am. J. Hum. Genet.* **91**, 1011–1021 (2012).

[5] Speed, D. & Evans, D. Estimating disease heritability from complex pedigrees allowing for ascertainment and covariates. *The American Journal of Human Genetics* **111**, 680–690 (2024).

[6] Mbatchou, J. *et al.* Computationally efficient whole-genome regression for quantitative and binary traits. *Nature genetics* **53**, 1097–1103 (2021).

[7] Jiang, L. *et al.* A resource-efficient tool for mixed model association analysis of large-scale data. *Nature genetics* **51**, 1749–1755 (2019).

[8] Yang, J., Zaitlen, N. A., Goddard, M. E., Visscher, P. M. & Price, A. L. Advantages and pitfalls in the application of mixed-model association methods. *Nature genetics* **46**, 100–106 (2014).

[9] Speed, D. *et al.* Reevaluation of SNP heritability in complex human traits. *Nat. Genet.* **49**, 986–992 (2017).

[10] Zeng, J. *et al.* Widespread signatures of natural selection across human complex traits and functional genomic categories. *Nature Communications* **12**, 1164 (2021).

[11] Schoech, A. P. *et al.* Quantification of frequency-dependent genetic architectures in 25 UK Biobank traits reveals action of negative selection. *Nature Communications* **10**, 790 (2019).

[12] Gazal, S. *et al.* Linkage disequilibrium-dependent architecture of human complex traits shows action of negative selection. *Nat. Genet.* **49** (2017).

[13] Zeng, J. *et al.* Signatures of negative selection in the genetic architecture of human complex traits. *Nat. Genet.* **50**, 746–753 (2018).

[14] Pazokitoroudi, A. *et al.* Efficient variance components analysis across millions of genomes. *Nature communications* **11**, 4020 (2020).

[15] Kang, H. M. *et al.* Efficient control of population structure in model organism association mapping. *Genetics* **178**, 1709–1723 (2008).

[16] Zhou, X. & Stephens, M. Genome-wide efficient mixed-model analysis for association studies. *Nature genetics* **44**, 821–824 (2012).

[17] Lippert, C. *et al.* Fast linear mixed models for genome-wide association studies. *Nature methods* **8**, 833–835 (2011).

[18] Svishcheva, G. R., Axenovich, T. I., Belonogova, N. M., Van Duijn, C. M. & Aulchenko, Y. S. Rapid variance components–based method for whole-genome association analysis. *Nature genetics* **44**, 1166–1170 (2012).

[19] Bulik-Sullivan, B. K. *et al.* LD Score Regression distinguishes confounding from polygenicity in genome-wide association studies. *Nature genetics* **47**, 291–295 (2015).

[20] Zhou, W. *et al.* Efficiently controlling for case-control imbalance and sample relatedness in large-scale genetic association studies. *Nature genetics* **50**, 1335–1341 (2018).

[21] Jiang, L., Zheng, Z., Fang, H. & Yang, J. A generalized linear mixed model association tool for biobank-scale data. *Nature Genetics* **53**, 1616–1621 (2021).

[22] Berrandou, T., Balding, D. & Speed, D. LDAK-GBAT: Fast and powerful gene-based association testing using summary statistics. *Am. J. Hum. Genet.* **110**, 23–29 (2023).

[23] Sudlow, C. *et al.* Uk Biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS medicine* **12**, e1001779 (2015).

[24] Bycroft, C. *et al.* The UK Biobank resource with deep phenotyping and genomic data. *Nature* **562**, 203–209 (2018).

[25] Hunter-Zinck, H. *et al.* Genotyping Array Design and Data Quality Control in the Million Veteran Program. *American Journal of Human Genetics* **106**, 535–548 (2020).

[26] Psychiatric GWAS Consortium Coordinating Committee. Genomewide association studies: History, rationale, and prospects for psychiatric disorders. *Am. J. Psychiatry* **166**, 540–556 (2009).

[27] de Leeuw, C. A., Mooij, J. M., Heskes, T. & Posthuma, D. MAGMA: generalized gene-set analysis of GWAS data. *PLoS computational biology* **11**, e1004219 (2015).

[28] Liu, Y. *et al.* Acat: A fast and powerful p-value combination method for rare-variant analysis in sequencing studies. *The American Journal of Human Genetics* **104**, 410–421 (2019).

[29] Kent, W. *et al.* The human genome browser at ucsc. *Genome Res.* **12**, 996–1006 (2002).

[30] Dey, R., Schmidt, E., Abecasis, G. & Lee, S. A fast and accurate algorithm to test for binary phenotypes and its application to phewas. *Am. J. Hum. Genet.* **101**, 37–49 (2017).

[31] Hastie, T., Tibshirani, R. & Friedman, J. *The Elements of Statistical Learning* (Springer, 2001).

[32] Privé, F., Aschard, H., Ziyatdinov, A. & Blum, M. Efficient analysis of large-scale genome-wide data with two R packages: bigstatsr and bigsnpr. *Bioinformatics* **34**, 2781–2787 (2018).

[33] Manichaikul, A. *et al.* Robust relationship inference in genome-wide association studies. *Bioinformatics* **26**, 2867–2873 (2010).

[34] Firth, D. Bias reduction of maximum likelihood estimates. *Biometrika* **80**, 27–38 (1993).

[35] Speed, D. & Balding, D. J. SumHer better estimates the snp heritability of complex traits from summary statistics. *Nature genetics* **51**, 277–284 (2019).