nature COMMUNICATIONS

ARTICLE

DOI: 10.1038/s41467-017-00470-2 OPEN

Non-parametric genetic prediction of complex traits with latent Dirichlet process regression models

Ping Zeng[1,2] & Xiang Zhou[2,3]

Using genotype data to perform accurate genetic prediction of complex traits can facilitate genomic selection in animal and plant breeding programs, and can aid in the development of personalized medicine in humans. Because most complex traits have a polygenic architecture, accurate genetic prediction often requires modeling all genetic variants together via polygenic methods. Here, we develop such a polygenic method, which we refer to as the latent Dirichlet process regression model. Dirichlet process regression is non-parametric in nature, relies on the Dirichlet process to flexibly and adaptively model the effect size distribution, and thus enjoys robust prediction performance across a broad spectrum of genetic architectures. We compare Dirichlet process regression with several commonly used prediction methods with simulations. We further apply Dirichlet process regression to predict gene expressions, to conduct PrediXcan based gene set test, to perform genomic selection of four traits in two species, and to predict eight complex traits in a human cohort.

[1] Department of Epidemiology and Biostatistics, Xuzhou Medical University, Xuzhou, Jiangsu 221004, China. [2] Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA. [3] Center for Statistical Genetics, University of Michigan, Ann Arbor, MI 48109, USA. Correspondence and requests for materials should be addressed to X.Z. (email: xzhousph@umich.edu)

Genome-wide association studies (GWASs) have identified thousands of genetic loci harboring associated single-nucleotide polymorphisms (SNPs) for many complex traits and diseases, providing unprecedented insights into the genetic basis of phenotypic variation[1–8]. The accumulation of genetic data from existing association studies has led to a growing interest in predicting traits and diseases using genetic markers (in addition to using traditional environmental or clinical variables)[9]. In animals or plants, accurate phenotype prediction with genetic markers can assist the selection of individuals with desirable breeding values and can improve the effectiveness of breeding programs[10]. In humans, accurate phenotype prediction with genetic markers can facilitate disease prevention and intervention at early stages and can aid in the development of personalized medicine by using genotype information to customize the treatment and predict the outcome[11]. Phenotype prediction has also been proposed recently as a key step for integrating functional genomic sequencing studies with GWASs: we can construct more powerful and interpretable gene-set tests in GWASs by setting variant weights to be the coefficients inferred from predictive models in expression quantitative trait locus mapping studies[12].

Progress toward accurate phenotype prediction requires the development of statistical methods that can model all SNPs jointly. Previous association studies have demonstrated that most complex traits and common diseases have a polygenic background and are each influenced by many genetic variants with small effects. For instance, it is estimated that thousands of causal variants influence human height[13]. Similarly, many animal or plant traits are contributed by hundreds of causal variants (e.g., maize-related traits, such as kernel oil and growing degree days (GDDs)[14, 15]; and cattle-related traits, such as backfat thickness, milk yield (MY) and hot carcass weight)[16, 17]. Because most complex traits and common diseases have a polygenic architecture, a handful of identified associated SNPs often only capture a small proportion of the phenotypic variation and thus cannot be used to yield accurate phenotype and risk prediction. Instead, accurate phenotype prediction requires polygenic models that can make use of all genome-wide SNPs[9, 18–20]. In the past decade, successful development and application of many polygenic models in the context of genomic selection has revolutionized many animal breeding programs[16, 21–23]. More recently, applications of polygenic models to human GWASs have also yielded fruitful results[11, 24–27].

Most existing polygenic models for prediction make an assumption on the effect size distribution and different methods differ mainly in such modeling assumption. For example, the commonly used linear mixed model (LMM), also known as the best linear unbiased predictor (BLUP), assumes that the effect sizes from all variants follow a normal distribution[9, 28]. The Bayes alphabetic (e.g., BayesA and BayesB) methods assume that the variant effect sizes follow a t-distribution or its variation[10, 18, 29]. The Bayesian lasso assumes a double exponential/Laplace distribution[30, 31]. NEG generalizes the Bayesian lasso by assuming a normal exponential gamma distribution[32]. BVSR and BayesCπ assume a point-normal distribution[29, 33]. BSLMM assumes a mixture of two normal distributions[34] and is closely related to the early reversible jump Markov Chain Monte Carlo (rjMCMC) method[20]. BayesR[35] assumes a three-component normal mixture together with a point mass at zero. Given the large number of modeling choices, one naturally wonders which method to use for any given trait. Previous studies have suggested that accurate prediction requires choosing a prior effect size distribution that can closely match the shape of the true effect size distribution,

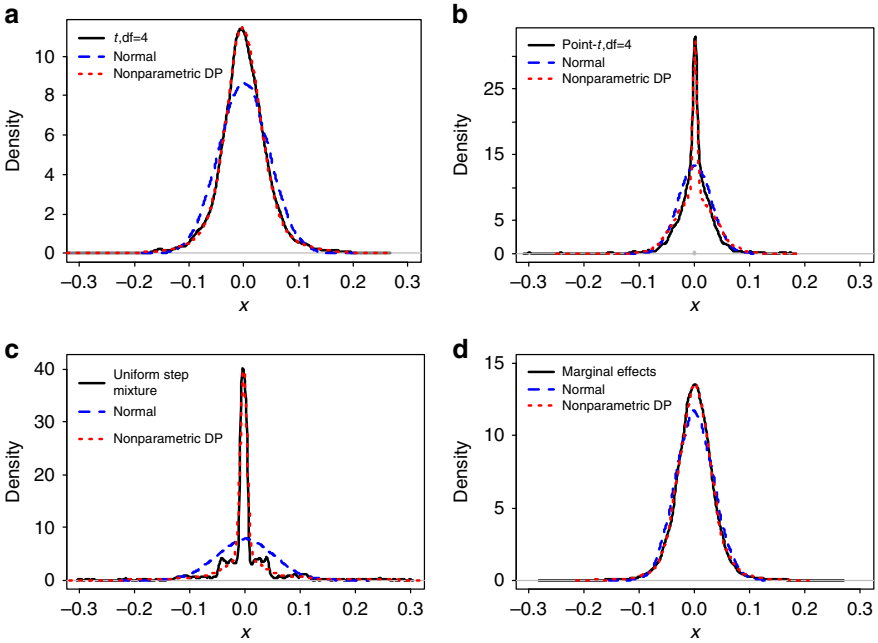


Fig. 1 The induced non-parametric Dirichlet process (nonparametric DP) normal mixture prior on the effect sizes can be used to approximate a large number of unimodal distributions. We either simulated 2000 values from **a** a standard t-distribution with df = 4; **b** a point-t mixture distribution with the zero proportion being 0.2, or equivalently, $0.8 \times t(\mathrm{df} = 4) + 0.2 \times \delta_0$, where δ_0 denotes a point mass at zero; **c** a four-component uniform step mixture distribution $0.50 \times U(-0.05,0.05) + 0.25 \times U(-0.3,0.3) + 0.15 \times U(-0.8,0.8) + 0.05 \times U(-2,2)$, where U denotes a uniform distribution; or obtained **d** the estimated marginal effect sizes from a linear mixed model in the cattle data with SCS (somatic cell score) as the phenotype. To make the first three data comparable with the last data in **d**, we centered and scaled the values from the first three data for them to have a mean of zero and within the range of (−0.3,0.3). We then fit each data with either our non-parametric distribution (*red*) or a normal distribution (*blue*), and displayed the fitted curves on top of the sample distribution (*black*). Clearly, the non-parametric Dirichlet process normal mixture can approximate all these distributions well, while a simple normal distribution cannot

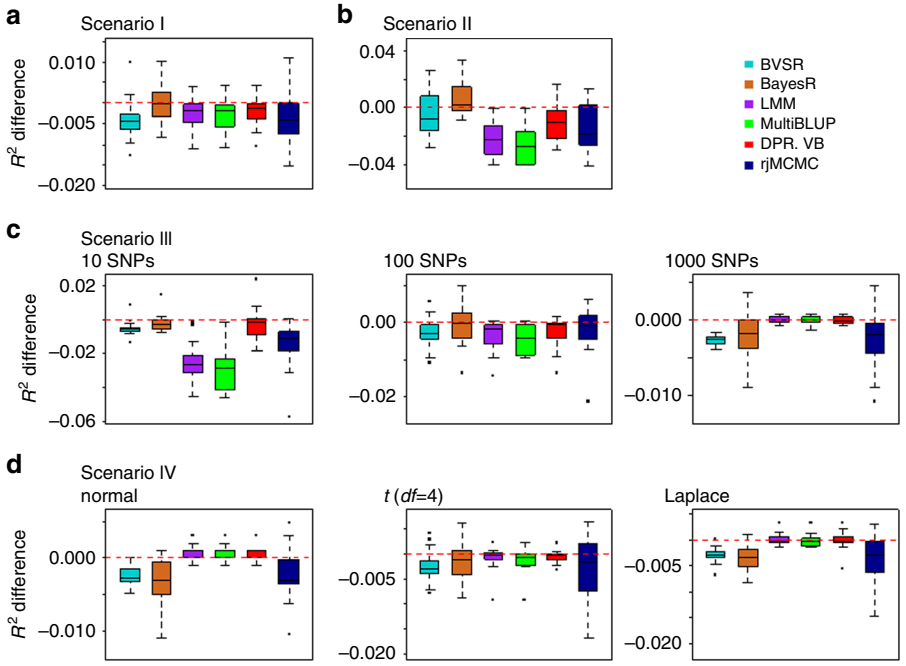


Fig. 2 Comparison of prediction performance of several methods with DPR.MCMC in simulations when PVE = 0.5. Performance is measured by R^2 difference with respect to DPR.MCMC, where a negative value (i.e., values below the *red horizontal line*) indicates worse performance than DPR.MCMC. The sample R^2 differences are obtained from 20 replicates in each scenario. Methods for comparison include BVSR (*cyan*), BayesR (*chocolate*), LMM (*purple*), MultiBLUP (*green*), DPR.VB (*red*), rjMCMC (*black blue*), and DPR.MCMC. Simulation scenarios include: **a** Scenario I, which satisfies the DPR modeling assumption; **b** Scenario II, which satisfies the BayesR modeling assumption; **c** Scenario III, where the number of SNPs in the large effect group is 10, 100, or 1000; and **d** Scenario IV, where the effect sizes are generated from either a normal distribution, a *t*-distribution or a Laplace distribution. For each *box plot*, the *bottom* and *top* of the box are the first and third quartiles, while the *ends of whiskers* represent either the lowest datum within 1.5 interquartile range of the lower quartile or the highest datum within 1.5 interquartile range of the upper quartile. For DPR.MCMC, the mean predictive R^2 in the test set and the standard deviation for the eight settings are, respectively, 0.272 (0.031), 0.299 (0.026), 0.295 (0.026), 0.281 (0.030), 0.277 (0.035), 0.278 (0.030), 0.282 (0.025), and 0.273 (0.022)

such that the inferred posterior can approximate well the polygenic architecture of the given trait[24, 35, 36]. However, the effect size distribution for any given trait or disease is unknown a priori and varies for different diseases in terms of the number of causal variants, their minor allele frequencies (MAFs), and their individual effect sizes[11]. Therefore, to achieve robust performance, it is important to design prior distributions that are flexible enough to resemble the true effect distribution in many traits as close as possible[34, 35].

Up to now, almost all existing polygenic models are parametric in nature and use a prior effect size distribution that is characterized by a few parameters. From the information channel perspective[37], the number of parameters in a parametric model determines model complexity and bounds the amount of information in data that can be captured by the model[37–40]. Therefore, using only a few parameters to characterize the effect size distribution can limit the flexibility of the model[37, 38] and impede its robust performance across a range of genetic architectures. As an example, the commonly applied LMM uses a normal distribution with one variance component parameter to characterize the effect size distribution. For highly polygenic traits, the assumed normal distribution can approximate the true effect size distribution well, and as a result, LMM can achieve good predictive performance[34, 35]. However, for traits with large effect variants, the assumed normal distribution can no longer capture the true effect size distribution well and the performance of LMM decays[34, 35].

To allow for greater flexibility on the a priori effect size distribution and to enable robust phenotype prediction performance across a range of phenotypes, we develop a Bayesian non-parametric model, which we refer to as the latent Dirichlet process regression (DPR). DPR does not use any fixed parametric distribution as the prior choice for the effect size distribution. Instead, DPR relies on the Dirichlet process to assign a prior on the effect size distribution itself and is thus capable of inferring an effect size distribution from the data at hand. Effectively, DPR uses infinitely many parameters a priori to character the effect size distribution, and with such a flexible modeling assumption, DPR is capable of adapting to a broad spectrum of genetic architectures and achieves robust predictive performance across a wide range of complex traits. We illustrate the benefits of DPR with simulations and real data applications for gene expression prediction, gene-based test via PrediXcan, genomic selection for four traits in two species, as well as genetic prediction of eight complex traits in a human cohort.

Results

Method overview. An overview of our method is provided in the Methods section with details provided in the Supplementary Note. Briefly, we use a Dirichlet process to introduce a non-parametric effect size distribution that can robustly resemble a large classes of unimodal distributions. Indeed, our prior effect size distribution can be used to adaptively and accurately approximate a t-distribution, a point-t mixture distribution, a mixture of step functions, as well as the marginal effect sizes estimated from a real data set; whereas a normal distribution cannot (Fig. 1). Therefore, our prior distribution on the effect size can adaptively approximate a wide range of possible effect size distributions underlying complex traits. Since accurate modeling

Table 1 Comparison of seven different methods in predicting gene expression levels in the GEUVADIS data

Threshold	ENET	BayesR	BVSR	LMM	MultiBLUP	rjMCMC	DPR	
							VB	MCMC
0.10	1061	809	486	**1195**	1098	1013	1163	**1280**
0.20	**449**	338	142	403	299	321	389	**467**
0.30	**182**	170	48	162	110	123	155	**194**
0.40	78	**84**	24	76	46	47	70	**86**
0.50	**37**	35	10	33	16	19	32	**38**
0.60	**15**	14	4	14	5	9	12	**17**
0.70	2	**3**	1	**3**	1	2	2	**3**

To compare prediction performance, we counted the number of genes whose median R^2 across 20 replicates in the test set is above a given R^2 threshold. A larger number thus indicates better performance. For each given threshold, we colored the best method with *red* and the second best method with *blue*

of the effect size distribution is a key to achieve accurate prediction performance[24, 34, 36], we expect our non-parametric model to perform robustly well across a range of polygenic architectures. Our method is implemented in the DPR software, freely available at http://www.xzlab.org/software.html.

Simulations. We first compare the performance of DPR with several other commonly used prediction methods using simulations. A total of seven different methods are included for comparison: (1) BVSR[29]; (2) BayesR[35]; (3) LMM[28]; (4) MultiBLUP[41]; (5) rjMCMC[20]; (6) DPR.VB, the variational Bayesian (VB) version of DPR; and (7) DPR.MCMC, the Markov chain Monte Carlo (MCMC) version of DPR. Note that both BayesR and MultiBLUP have been recently demonstrated to outperform a range of existing prediction methods; thus, we do not include other prediction methods into comparison here.

To make our simulations as real as possible, we used genotypes from an existing cattle GWAS data set[17] with 5024 individuals and 42,551 SNPs and simulated phenotypes. To cover a range of possible genetic architectures, we consider eight simulation settings from four different simulation scenarios with the phenotypic variance explained (PVE) by all SNPs being either 0.2, 0.5, or 0.8 (details in Methods). In each setting for each PVE value, we performed 20 simulation replicates. In each replicate, we randomly split the data into a training data with 80% individuals and a test data with the remaining 20% individuals. We then fitted different methods on the training data and evaluated their prediction performance on the test data (i.e., Monte Carlo cross validation). We evaluated prediction performance using either the squared correlation coefficient (R^2) or mean squared error (MSE). We contrasted the prediction performance of all other methods with that of DPR.MCMC by taking the difference of R^2 or MSE between the other methods and DPR.MCMC. Therefore, an R^2 difference below zero or an MSE difference above zero suggests worse performance than DPR.MCMC. Figure 2 shows R^2 differences for different methods across 20 replicates in each of the eight simulation settings for PVE = 0.5. Because Fig. 2 shows prediction performance difference, a large sample variance of a method in the figure only implies that the prediction performance of the method differs a lot from that of DPR.MCMC, but does not imply that the method itself has a large variation in predictive performance. Supplementary Table 1 shows the standard deviation of absolute R^2 values across cross variation replicates; various methods display similar prediction variability. Supplementary Figs. 1 and 2 show the R^2 differences for PVE = 0.2 and PVE = 0.8, respectively. The corresponding results for MSE differences are shown in Supplementary Figs. 3–5. The R^2 and MSE values of the baseline method, DPR.MCMC, are shown in the corresponding figure legend.

Overall, while each method works the best when their individual modeling assumption is satisfied, DPR.MCMC is robust and works well across all eight settings from four scenarios. For example, if we rank the methods based on their median performance across replicates, then when the total PVE is moderate (e.g., PVE = 0.5, Fig. 2; note that for each PVE there are a total of eight simulation settings for the four scenarios), DPR.MCMC is the best or among the best in seven simulation settings (i.e., scenario I, $c = 10$, 100, and 1000 in scenario III, and normal, t and Laplace distributions in scenario IV; where "among the best" refers to the case when the difference between the given method and the best method is within ± 0.001) and is ranked as the second best in the rest one simulation setting (i.e., scenario II). Similarly, when the total PVE is high (e.g., PVE = 0.8, Supplementary Fig. 2), DPR.MCMC is the best or among the best in seven simulation settings, and it is ranked as the second best in scenario IV when the effect size follows a normal distribution. Even when DPR.MCMC is ranked as the second best method, the difference between DPR.MCMC and the best method is often small. Among the rest of the methods, LMM, MultiBLUP, and rjMCMC all work well in polygenic settings (scenario I; $c = 1000$ in scenario III; scenario IV) but can perform poorly in sparse settings (scenario II; $c = 10$ and $c = 100$ in scenario III). The performance of LMM, MultiBLUP, and rjMCMC in polygenic vs. sparse settings presumably stems from their polygenic assumptions on the effect size distribution. In contrast, because of the sparse assumption on the effect size distribution, both BayesR and BVSR have an advantage in sparse settings (scenario II; $c = 10$ or 100 in scenario III) but suffers in polygenic settings ($c = 1000$ in scenario III; scenario IV). The performance of BVSR is also generally worse than BayesR in the challenging setting when PVE is either small or moderate, presumably because of the much simpler prior assumption employed in BVSR for the non-zero effects. Finally, the VB version of DPR (i.e., DPR.VB) performs considerably less well compared with the MCMC version of DPR (i.e., DPR.MCMC), especially when PVE is high (Supplementary Fig. 2). However, DPR.VB still compares favorably with the other methods when PVE is small or moderate (Supplementary Fig. 1).

Real data applications. To gain further insights, we compare the performance of DPR with the other methods in several real data sets to (1) predict gene expression levels using cis-SNPs; (2) conduct subsequent PrediXcan based gene set test; (3) perform genomic selection in animal studies; and (4) predict complex traits in humans.

Our first application is predicting gene expression levels using cis-SNPs in the GEUVADIS data[42]. The GEUVADIS data contains gene expression measurements on 15,810 genes and 465 individuals after quality control (Methods). These individuals

Table 2 Comparison of seven different methods in the PrediXcan gene set test in the WTCCC data

Disease	ENET	BayesR	BVSR	LMM	MultiBLUP	rjMCMC	DPR	
							VB	MCMC
T1D	21	22	16	23	22	24	**26**	**25**
CD	**6**	0	1	4	4	**5**	3	**6**
RA	7	1	5	**9**	**8**	7	**8**	7
BD	0	0	0	0	0	0	0	0
CAD	0	0	0	0	0	0	0	0
HT	0	0	0	0	0	0	0	0
T2D	0	0	0	0	0	0	0	0
Total	34	23	22	36	34	36	**37**	**38**

The table lists the number of genes passing the genome-wide significance threshold via Bonferroni correction ($\alpha = 1.15 \times 10^{-5}$) in each of the seven common diseases. For each disease, we colored the best method with *red* and the second best method with *blue*

have their genotypes measured in the 1000 Genomes project[43]. In the data, we first identified cis-SNPs that are within 100 kb of each gene and obtained an average of 175 cis-SNPs per gene. Then, for each gene in turn, we applied different methods to predict gene expression levels using these cis-SNPs. To measure prediction performance, we carried out 20 Monte Carlo cross validation data splits as in simulations. In each data split, we fitted methods in a training set with 80% of randomly selected individuals and evaluated method performance using R^2 in the test set with the remaining 20% of individuals. In addition to the seven methods used in the simulations (i.e., LMM, BVSR, MultiBLUP, BayesR, rjMCMC, DPR.VB, and DPR.MCMC), we also applied Elastic Net (ENET)[44], which is the default method used in the original PrediXcan paper[12]. Table 1 lists the number of genes with a predictive R^2 above different thresholds for different methods. The predictive R^2 obtained from DPR.MCMC vs. various other methods across all genes is shown as scatter plots in Supplementary Fig. 6, where each plot also lists the number of genes for which DPR.MCMC performs better and the number of genes for which DPR.MCMC performs worse.

The results are largely consistent with these in simulations. Overall, DPR.MCMC generally achieves better predictive performance than the other methods. For example, DPR.MCMC is able to achieve a higher predictive $R^2 > 0.10$ in ~1300 genes, which is ~100 more than that by the second best method at this threshold (i.e., LMM; Table 1). Similarly, compared with other methods, not only does DPR.MCMC achieve a higher R^2 for most genes; the R^2 improvement from DPR.MCMC can be large for many genes (Supplementary Fig. 6). Among the rest of the methods, the performance of LMM, DPR.VB, and ENET are comparable with each other and are ranked right behind DPR.MCMC. On the other hand, the two sparse models (i.e., BVSR and BayesR) perform poorly in this data, especially for some genes whose expression levels are highly predictive by the other methods (Table 1, Supplementary Fig. 6).

The robust performance of DPR.MCMC in predicting gene expression levels also translates to a relatively high power in the subsequent PrediXcan gene set test. To perform PrediXcan gene set test, we consider the seven common diseases from Wellcome Trust Case Control Consortium (WTCCC)[4] as in Gamazon et al.[12]. For each disease and each gene in turn, we used the estimated cis-SNP effect sizes on expression levels from GEUVADIS as weights to construct gene set tests in WTCCC. Following Gamazon et al.[12], we focused on a set of 4343 genes that had a predictive R^2 above 0.01 from all methods. The results are shown in Table 2, which lists the number of significant genes identified by different methods for different diseases. In total, DPR.MCMC identified 38 genes associated with different phenotypes, more than that identified by any other methods. The performance of DPR.MCMC is followed by DPR.VB and subsequently LMM and rjMCMC. Supplementary Table 2 lists the significant genes identified by DPR.MCMC, which are all consistent with previous studies. We also note that, in general, a higher gene expression predictive performance leads to a higher power in the subsequent gene set analysis. In addition, consistent with their relatively poor gene expression prediction performance, the two sparse models (BayesR and BVSR) do not perform well in the gene set test as well.

Finally, we compare the performance of DPR with the other methods in predicting phenotypes in three GWAS data sets: (1) a cattle study[17], where we focus on three phenotypes: milk fat percentage (MFP), MY, as well as somatic cell score (SCS); (2) a maize study[15], where we use GDD as the phenotype; (3) the Framingham heart study (FHS) data[45], where we focus on five plasma traits that include low-density lipoprotein (LDL) cholesterol, glucose (GLU), high-density lipoprotein (HDL) cholesterol, total cholesterol (TC) and triglycerides (TGs), and three anthropometric traits that include height, weight and body mass index (BMI). As in simulations, for each phenotype, we performed 20 Monte Carlo cross validation data splits. In each data split, we fitted methods in a training set with 80% of randomly selected individuals and evaluated method performance using R^2 or MSE in a test set with the remaining 20% of individuals. We again contrasted the performance of the other methods with that of DPR.MCMC by taking the R^2 difference or MSE difference with respect to DPR.MCMC. The results are shown in Fig. 3 (R^2 difference) and Supplementary Fig. 7 (MSE difference), with R^2 and MSE of DPR.MCMC presented in the corresponding figure legend. Supplementary Table 1 shows the standard deviation of absolute R^2 values across cross variation replicates. Supplementary Fig. 8 also displays trace plots of the log posterior likelihood from DPR.MCMC for all traits, suggesting reasonable convergence of our method.

Overall, consistent with simulations, DPR.MCMC shows robust performance across all traits and is ranked either as the best or the second best method. In the cattle data (Fig. 3a), for MFP and MY, both DPR.MCMC and BayesR perform the best. For SCS, DPR.MCMC performs the best, followed by BayesR. rjMCMC performs well for MFP and MY but poorly for SCS; while LMM and MultiBLUP do not perform well for MFP and MY in the cattle data but their performance improves for SCS. The relative performance of BayesR vs. LMM and MultiBLUP in the cattle data is consistent with the distinct genetic architectures that underlie the three complex traits[17, 46]: while MFP and MY are affected by a few large or moderate effect SNPs together with many small effect SNPs, SCS is a highly polygenic trait influenced by many SNPs with small effects. BVSR performs poorly for these three traits in the cattle data. In the maize data (Fig. 3a), DPR.MCMC performs the best,

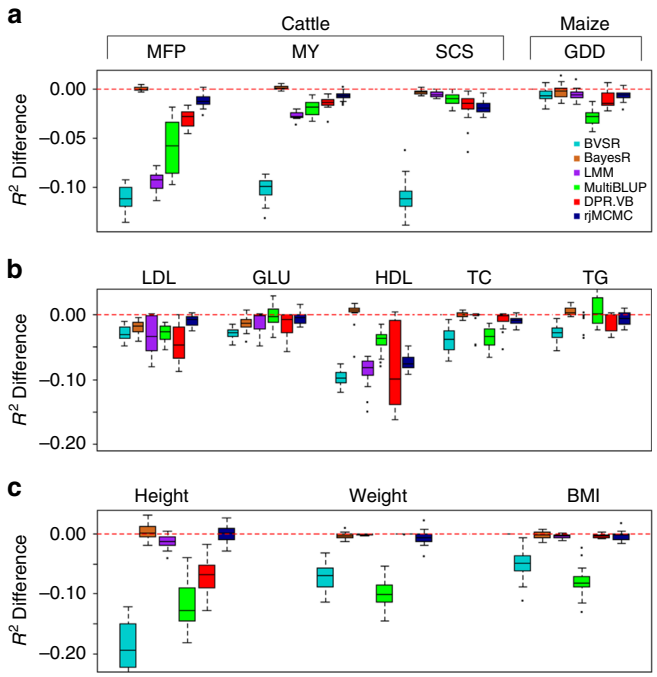


Fig. 3 Comparison of prediction performance of several methods with DPR.MCMC for twelve traits from three data sets. **a** Prediction performance for MFP, MY and SCS in the cattle data, and for GDD in the maize data; **b** Prediction performance for the five plasma traits in the FHS data; c Prediction performance for the three anthropometric traits in the FHS data. Performance is measured by R^2 difference with respect to DPR.MCMC, where a negative value (i.e., values below the *red horizontal line*) indicates worse performance than DPR.MCMC. Methods for comparison include BVSR (*cyan*), BayesR (*chocolate*), LMM (*purple*), MultiBLUP (*green*), DPR.VB (*red*), rjMCMC (*black blue*), and DPR.MCMC. For each *box plot*, the *bottom* and *top* of the box are the first and third quartiles, while the *ends of whiskers* represent either the lowest datum within 1.5 interquartile range of the lower quartile or the highest datum within 1.5 interquartile range of the upper quartile. The sample R^2 differences are obtained from 20 replicates of Monte Carlo cross validation for each trait. For DPR.MCMC, the mean predictive R^2 in the test set and the standard deviation across replicates are 0.751 (0.011) for MFP, 0.624 (0.012) for MY, 0.551 (0.017) for SCS and 0.828 (0.012) for GDD, 0.081 (0.033) for LDL, 0.047 (0.017) for GLU, 0.153 (0.044) for HDL, 0.050 (0.020) for TC, 0.044 (0.015) for TG, 0.478 (0.031) for height, 0.169 (0.038) for weight, and 0.137 (0.037) for BMI. The SNP heritability estimates are 0.912 (0.007) for MFP, 0.810 (0.012) for MY, 0.801 (0.012) for SCS, 0.880 (0.013) for GDD, 0.397 (0.024) for LDL, 0.357 (0.036) for GLU, 0.418 (0.024) for HDL, 0.402 (0.036) for TC, 0.334 (0.034) for TG, 0.905 (0.013) for height, 0.548 (0.022) for weight and 0.483 (0.023) for BMI

followed by BayesR, suggesting that GDD is influenced by a few SNPs with large effects[15]. In the FHS data (Fig. 3b, c), DPR.MCMC performs the best or among the best for LDL, GLU, TC, Weight, and BMI. Its performance is comparable to BayesR and rjMCMC for Height, and follows right behind BayesR for HDL and TG. Besides DPR and BayesR, rjMCMC also performs well in FHS and is often ranked as the third best method (e.g., for LDL, GLU, Height, and Weight). In contrast to the relatively robust performance of DPR.MCMC, however, all other methods can perform poorly for certain phenotypes. In Fig. 3, for example, BayesR is the second worst method for predicting GLU; LMM is the second worst method for predicting LDL; MultiBLUP is the worst method for predicting Weight and BMI; DPR.VB performs among the worst for LDL and HDL; rjMCMC performs poorly for HDL; while BVSR performs the worst for almost all traits except for LDL, Height, and Weight. The poor performance of BVSR presumably stems from its relatively simple and sparse assumption on the effect sizes.

Because the FHS is a family based study, we use this data to further examine the influence of individual relatedness on prediction performance. To do so, we divided the FHS data into two sub data sets (D1 and D2) with equal sample size but different levels of relatedness (details in Methods): individuals in D1 are more closely related to each other than those in D2. We then compared method performance by performing cross validation in each of the two data sets separately. While the difference between methods becomes smaller due to smaller sample size in the two sub data sets, the relative performance of most methods for the eight traits are largely unchanged in these two sub data sets as compared to that in the complete data (Supplementary Fig. 9 vs. Fig. 3b, c). For example, DPR.MCMC is ranked as the best method or among the best methods for six traits in D1 and for four traits in D2. BayesR performs similarly and is ranked as the best or among the best for four traits in D1 and for five traits in D2. LMM ranks right after DPR.MCMC and BayesR, while the other methods do not perform well here. In addition, all methods generally perform better in D1 than in D2 (Supplementary Fig. 10), suggesting that relatedness improves prediction performance—a phenomenon that has been well recognized by previous studies[9, 23, 47–49]. Besides cross validation within each data set separately, we also performed cross validation between D1 and D2 by predicting traits in one data with parameters inferred from another. The results are again largely consistent with the main results. In particular, DPR.MCMC is ranked as the best or among the best for six traits in D1 to D2 prediction and for eight traits in D2 to D1 prediction. BayesR is ranked as the best or among the best for six traits in D1 to D2 prediction and for eight traits in D2 to D1 prediction. rjMCMC also performs reasonably well and follows right behind DPR.MCMC and BayesR (Supplementary Fig. 11).

Computational time. Finally, we list the computational time of the seven methods for the 12 traits in Table 3. Note that some differences in computational time among methods may reflect implementation issues, including the language environment in which the methods are implemented, rather than fundamental differences between algorithms. In addition, we only list in the table the computation time in the fitting stage. Computation time spent in the prediction stage by plugging in estimated coefficients in the new data is almost ignorable and is thus not listed. For sampling based methods (BVSR, rjMCMC, DPR.MCMC, and BayesR), we measure the computational time based on a fixed number of iterations. However, due to different convergence properties of different algorithms (e.g., BVSR uses a Metropolis-Hastings algorithm, rjMCMC uses a reversible jump MCMC algorithm, while both DPR.MCMC and BayesR use a Gibbs sampling algorithm), a fixed number of iterations in different methods may correspond to different mixing performance. Nevertheless, we can see that DPR.MCMC has a similar computational cost as the other Gibbs based approach (e.g., BayesR), though in the human data both these Gibbs based approaches (DPR.MCMC and BayesR) can be slower than the Metropolis-Hastings approach (BVSR) and the reversible jump MCMC algorithm (rjMCMC) that effectively update only a small subset of significant SNPs in each iteration. In contrast, DPR.VB is orders of magnitude faster than its MCMC counterpart, and is computationally as efficient as the other two non-MCMC based approaches (LMM and MultiBLUP).

Table 3 Mean computational time of the seven methods in the model fitting stage for 12 traits across three data sets

Data	Traits	BVSR	rjMCMC	BayesR	LMM	MultiBLUP	DPR	
							VB	MCMC
Cattle	MFP	2.26 (0.49)	3.04 (0.24)	5.01 (0.75)	0.27 (0.05)	0.40 (0.12)	0.22 (0.11)	6.29 (3.07)
	MY	2.51 (0.52)	2.95 (0.31)	5.95 (1.04)	0.27 (0.08)	0.46 (0.07)	0.21 (0.09)	4.01 (0.55)
	SCS	4.56 (0.78)	3.15 (0.27)	6.17 (1.05)	0.24 (0.04)	0.27 (0.06)	0.20 (0.08)	5.23 (2.38)
Maize	GDD	2.38 (0.72)	1.08 (0.11)	7.86 (1.57)	0.19 (0.05)	0.03 (0.01)	0.08 (0.01)	4.53 (1.29)
FHS	LDL	1.02 (0.17)	1.78 (0.15)	78.56 (27.78)	1.76 (1.15)	1.71 (0.33)	1.24 (0.79)	85.76 (18.22)
	GLU	0.25 (0.14)	1.86 (0.18)	47.87 (17.86)	1.06 (0.52)	1.63 (0.13)	0.43 (0.12)	61.16 (23.46)
	HDL	0.49 (0.16)	1.83 (0.14)	80.45 (38.23)	3.39 (1.26)	1.74 (0.11)	1.28 (0.56)	84.38 (10.61)
	TC	0.24 (0.13)	1.92 (0.12)	51.17 (16.72)	1.05 (0.48)	1.62 (0.37)	0.42 (0.11)	51.69 (11.77)
	TG	0.25 (0.17)	1.98 (0.15)	59.41 (17.72)	0.99 (0.35)	1.91 (0.46)	0.45 (0.13)	50.78 (10.72)
	Height	0.68 (0.16)	1.75 (0.16)	71.14 (13.80)	2.27 (1.12)	4.13 (1.18)	1.56 (0.18)	71.62 (11.89)
	Weight	0.59 (0.13)	1.61 (0.15)	72.66 (12.15)	2.28 (1.11)	1.95 (0.34)	1.61 (0.10)	79.67 (15.04)
	BMI	0.47 (0.10)	1.71 (0.13)	76.08 (15.28)	2.31 (1.13)	2.35 (0.27)	1.57 (0.17)	75.15 (14.91)

The computational time is in hours. Values in parentheses are standard deviations. Mean and standard deviation are calculated based on 20 replicates. For MCMC-based methods (rjMCMC, BVSR, BayesR, and DPR.MCMC), the computational time is based on 50,000 iterations of Metropolis Hastings steps for BVSR, reversible jump steps for rjMCMC, and Gibbs steps for BayesR and DPR.MCMC

Discussion

We have presented a novel statistical method, DPR, for genetic prediction of complex traits. DPR uses an infinitely many parameters a priori to flexibly model the effect size distribution, and represents the first non-parametric method developed for modeling polygenic traits in genetic association studies. By flexibly modeling the effect size distribution, DPR is capable of adapting to the polygenic architecture underlying many complex traits and enjoys robust performance across a range of phenotypes. With simulations and applications to four real data sets, we have illustrated the benefits of DPR.

We have focused on one application of DPR—genetic prediction of phenotypes. Like some other polygenic methods[34, 35, 50], DPR can also be applied to many other polygenic applications. For example, DPR can be used to estimate the proportion of variance in phenotypes explained by all SNPs, a quantity that is commonly referred to as SNP heritability[28, 34]. Because DPR assumes a flexible effect size distribution that is adaptive to the genetic architecture underlying a given trait, it has the potential to provide accurate estimation of SNP heritability. As another example, DPR can be applied to association mapping. There, we can view the normal component with the smallest variance as the polygenic background, and we can estimate the probability of a SNP being in any normal components other than the smallest one as the posterior inclusion probability (PIP). PIP computed in this way measures SNP marginal association strength in the presence of polygenic effects, and may represent a more powerful association indicator than standard single SNP association test statistics[33, 50]. An important feature of using PIP in the context of Bayesian models is that PIP quantifies the uncertainty of association strength[33, 50], which is a desirable feature that is not easily achieved by penalized frequentist counterparts[51].

Here, we have restricted ourselves to applying DPR to continuous phenotypes. For case control studies, we could follow previous approaches of treating binary phenotypes as continuous traits and apply DPR directly[34, 35, 41]. However, it would be desirable to extend DPR to accommodate case control data or other discrete data types in a principled way, by, for example, extending DPR into the generalized linear model framework. In particular, we could use a probit or a logistic link to extend DPR to directly model case control data. We could use a Poisson or an over-dispersed Poisson distribution to extend DPR to model count data. Extending DPR to various discrete data types would likely lead to wider applications of DPR beyond GWASs. For instance, by modeling count data, DPR could be used to perform differential expression analysis[52] or expression QTL mapping in RNA sequencing studies[53, 54]. Similarly, by modeling proportional data, DPR could be used to perform differential methylation analysis or methyl-QTL mapping in bisulfite sequencing studies[55]. Extending DPR to modeling discrete data types using the generalized linear model framework is thus an important avenue for future research.

In the present study, while we used unrelated individuals for GEUVADIS gene expression prediction and PrediXcan tests, we used related individuals for the other two real data applications. Related individuals not only share similar genetic background but also are likely influenced by a common set of environmental factors[47, 56]. In addition, untyped causal SNPs in related individuals can be more easily tagged by neighboring typed SNPs than that in unrelated individuals, thanks to the relatively high linkage disequilibrium (LD) in related data. Because both the shared environmental factors and easy tagging of causal SNPs can facilitate prediction, cross validation using related individuals often results in better prediction performance than using unrelated individuals[9, 23, 47–49]. However, we caution that the prediction accuracy measured in the test data obtained with cross-validation in related individuals are likely inflated if our ultimate goal is to perform prediction in unrelated individuals instead of related ones. In addition, the predictive model inferred from related individuals may not generalize well to unrelated individuals who are not necessarily influenced by the same set of environmental factors and who do not share the same LD pattern near the causal SNPs. We have attempted to tease apart the influence of relatedness on prediction performance by splitting the FHS data into two parts with different levels of relatedness. Our results indeed show that, while the relative performance of various methods remains largely the same, the absolute performance of all methods do increase with individual relatedness. Additionally, while our method performs well relative to the other methods, we caution that DPR's prediction accuracy is still unlikely of practical use in human clinical setting. Studies on unrelated individuals or studies using a fully independent validation data are likely required to establish the practical utility of prediction methods, which often have unsatisfactory performance there[9, 47, 57]. Despite the practical importance of using completely independent or cross-population studies for prediction performance validation, however, we also want to point out its potential caveat: using completely independent data for cross-validation may fail to correctly characterize the relative performance of different methods. In particular, a good method that properly captures the signal in the training data may suffer in the validation data due to different LD patterns between the two

data sets. Similarly, a poor method that fails to capture the signal in the training data may perform well in the validation data where such signal is no longer relevant. Therefore, using training and validating data that are both representative of the study population is important to not only ensure a proper comparison among methods but also to ensure the clinical relevance and wide applicability of the prediction methods. Exploring the use of such data is an important direction for future research.

DPR is not without its limitations. Perhaps the biggest limitation is its computational cost. Like any other MCMC-based approaches[34, 35, 58], our Gibbs algorithm for fitting DPR is computationally slow and can only be applied to moderate-sized GWAS studies. To make DPR widely applicable, we have explored the use of VB approximation for fitting DPR. VB approximation obtains an approximate posterior distribution through optimization[59] and represents a much faster alternative to MCMC sampling. Indeed, DPR.VB is orders of magnitude faster than DPR.MCMC. However, despite its faster computational speed, the VB algorithm is less accurate than MCMC when SNP heritability is large, sometimes by quite a large margin (e.g., PVE = 0.80 in simulations). The loss of accuracy in VB is not unexpected because our VB assumes that the posterior distributions of the SNP effect sizes are independent from each other. Posterior independence is an unrealistic assumption given that SNP genotypes are correlated through LD. Therefore, it is important to explore alternative VB algorithms to incorporate the posterior correlation among effect sizes, by, for example, adapting algorithms developed elsewhere[60, 61]. It would be ideal if we could develop algorithms that can achieve a high predictive performance as DPR.MCMC but incurs a small computational cost as DPR.VB. Certainly, besides developing alternative algorithms to MCMC, there is still room for improvement on our MCMC algorithm. For example, we could use all individuals to compute some quantities while use only a subset of individuals to compute other quantities, as in our previous MQS method[62], in order to reduce the computational burden while maintaining the accuracy of the algorithm. In any case, developing efficient and accurate algorithms likely represents a key step to adapt existing polygenic methods to association studies that are orders of magnitude larger.

Methods

Overview of DPR. We provide a brief overview of DPR here. Detailed methods and algorithms are provided in the Supplementary Note. To model the relationship between phenotypes and genotypes, we consider the following multiple regression model

$$\mathbf{y} = \mathbf{W}\alpha + \mathbf{X}\beta + \epsilon,\ \epsilon \sim N\left(0, \sigma_e^2 I_n\right), \tag{1}$$

where $\mathbf{y}$ is an n-vector of phenotypes measured on n individuals; $\mathbf{W}$ is an n by c matrix of covariates including a column of 1 s for the intercept term; $\boldsymbol{\alpha}$ is a c-vector of coefficients; $\mathbf{X}$ is an n by p matrix of genotypes; $\boldsymbol{\beta}$ is the corresponding p-vector of effect sizes; $\boldsymbol{\epsilon}$ is an n-vector of residual errors where each element is assumed to be independently and identically distributed from a normal distribution with variance σ_e^2.

Like many previous methods[9, 19, 28, 34, 41], we assume that the effect size of ith SNP, β_i, follows a normal distribution with variance σ^2, i.e., $\beta_i \sim N(0, \sigma^2)$. Unlike previous methods, however, we specify a non-parameter prior on the hyper-parameter σ^2 to induce a non-parametric prior on β_i. To motivate our prior choice for σ^2, it helps to provide a brief review of the previous polygenic prediction methods. Among the many polygenic prediction methods developed recently, a surprisingly large number of them assume a priori that the effect sizes follow a particular class of prior distribution—the scale mixture of normal distributions[63]. Specifically, these methods assume that each effect size β_i follows a normal distribution $\beta_i \sim N(0, \sigma^2)$, with the variance parameter (i.e., the scale parameter) σ^2 following another distribution $p(\sigma^2)$. The prior distribution on σ^2, $p(\sigma^2)$, thus differentiates many different predictive methods. For example, LMM assumes a flat prior $p(\sigma^2)$ that is proportional to a constant[9, 28]. The Bayes alphabetic methods assume that σ^2 follows an inverse gamma distribution to induce a t-prior on β_i[10, 18, 64]. The Bayesian lasso assumes that σ^2 follows a Rayleigh distribution to induce a double exponential distribution (a.k.a. Laplace distribution) on β_i[30, 58]. NEG assumes an exponential gamma distribution on σ^2 to induce an NEG prior on β_i[32]. BVSR and BayesCπ assume a mixture of a point mass at zero with another flat prior to induce a point-normal distribution on β_i[29, 33]. BSLMM assumes a mixture of two point masses to induce a normal mixture distribution on β_i[34]. While BayesR assumes a three point masses together with another point mass at zero on σ^2 to also induce a normal mixture distribution on β_i[35].

The scale mixture of normal distributions is flexible because different distributions on the scale parameter σ^2 can be used to induce many smooth unimodal distributions on β_i. However, existing predictive methods explicitly make a parametric prior assumption on σ^2, which necessarily relies on a limited number of parameters to characterize the distribution on σ^2. Consequently, the induced effect size distribution on β_i from a parametric prior on σ^2 can be restrictive and may sometimes fail to resemble closely the unknown truth effect size distribution underlying complex traits. Motivated by the potential drawback of parametric priors on σ^2, we instead develop a non-parametric prior distribution on σ^2 to induce a more flexible distribution on β_i. Because a non-parametric distribution is characterized by effectively infinitely many parameters, our induced effect size distribution on β_i has the potential to resemble a wide range of genetic architectures and achieve robust predictive performance across a variety of traits.

Technically, we assume σ^2 follows a Dirichlet process (DP) prior[37–40]

$$\sigma^2 \sim G, G \sim \mathrm{DP}(H, \lambda), \tag{2}$$

where H is the base distribution, and λ is the concentration parameter that describes how the distribution on σ^2, G, deviates from the base distribution. Here, we use an inverse gamma distribution as the base distribution and set the two parameters in the inverse gamma distribution to small values to keep the prior relatively uninformative. We treat the concentration parameter λ as an unknown hyper-parameter and intend to infer it from the data at hand. Because we use the Dirichlet process as a prior for the latent variance parameter σ^2 we refer to our regression model based on Eqs. (1) and (2) as the latent Dirichlet Process Regression, or DPR. The induced marginal distribution on β_i (after integrating out σ^2) is also non-parametric and can robustly resemble a large classes of unimodal distributions. Indeed, the distribution on β_i can be used to adaptively and accurately approximate a t-distribution, a point-t mixture distribution, a mixture of step functions, as well as the marginal effect sizes estimated from a real data set; whereas a normal distribution cannot (Fig. 1). Therefore, our prior distribution on the effect size can adaptively approximate a wide range of possible effect size distributions underlying complex traits. Since accurate modeling of the effect size distribution is a key to achieve accurate prediction performance[24, 34, 36], we expect our non-parametric model to perform robustly well across a range of polygenic architectures.

It is important to point out that our modeling assumption on the effect sizes β_i is also mathematically equivalent to a Dirichlet process normal mixture, which is a mixture of normal distributions with infinitely many normal components. Specifically, using the stick-breaking constructive representation of the Dirichlet process[59], we can re-write our modeling assumption on β_i in an equivalent form as

$$\begin{aligned} \beta_i &\sim \sum_{k=1}^{+\infty} \pi_k N\left(0, \sigma_k^2\right), \\ \pi_k &= \nu_k \prod_{l=1}^{k-1} (1-\nu_l), \nu_k \sim \mathrm{Beta}(1, \lambda), \end{aligned} \tag{3}$$

where λ is the same concentration parameter as in Eq. (2), and determines the number of normal components in the model and subsequently the model complexity[59]. Each σ_k^2 in the above equation follows the base distribution H. From the normal mixture equivalence aspect, our method effectively generalizes many previous methods[18, 34, 35] that use a fixed, often small, number of normal components to using infinitely many normal components a priori. Although the prior number of normal components in our model is infinite, the posterior number of components for any given data set will be finite, and can be automatically inferred based on the data at hand. Therefore, our model has the potential to adjust the model complexity according to the data complexity, and has the potential to adapt to a wide range of polygenic architectures.

To fit our model, we develop two complementary algorithms: one is based on the MCMC algorithm, and the other is based on the VB approximation. The MCMC sampling algorithm, which we refer to as DPR.MCMC, is accurate but computationally slow. The VB algorithm, which we refer to as DPR.VB, is computationally fast, but, as we will show in the results, is often less accurate. The two algorithms provide users the choice of speed vs. accuracy. The details of the two algorithms are provided in Supplementary Note.

Simulations. We used genotypes from an existing cattle GWAS data set[17] with 5024 individuals and 42,551 SNPs and simulated phenotypes. To cover a range of possible genetic architectures, we consider *eight* simulation settings from four different simulation scenarios to cover a range of possible genetic architectures:

Scenario I satisfies the DPR modeling assumption, where all SNPs are causal and SNPs in different effect-size groups have different effects. Specifically, we randomly selected 10 group-one SNPs, 100 group-two SNPs, 1000 group-three

SNPs, and set the remaining SNPs as group-four SNPs. We simulated SNP effect sizes all from a standard normal distribution but scaled their effects in each group separately so that the proportion of genetic variance explained by the four groups are 0.05, 0.15, 0.20, and 0.60, respectively. We set the total proportion of phenotypic variance (PVE; i.e., SNP heritability) to be either 0.2, 0.5, or 0.8, representing low, moderate, and high heritability, respectively. This simulation scenario consists of one simulation setting for each PVE.

Scenario II satisfies the BayesR modeling assumption, where a small proportion of SNPs are causal. These causal SNPs come from three effect-size groups. The simulations were similar to scenario I with the only exception that the group-four SNPs have zero effects. Here, the proportion of PVE by the three groups are 0.1, 0.2, and 0.7, respectively. Again, we set the total PVE to be either 0.2, 0.5, or 0.8. This simulation scenario consists of one simulation setting for each PVE.

Scenario III is similar to Scenario I except that SNPs come from two effect-size groups, thus representing a simpler scenario than I. In particular, we selected either $c = 10$, 100, or 1000 SNPs as group-one SNPs and set the remaining SNPs as group-two SNPs. We simulated their effect sizes from a standard normal distribution and scaled their effects in each group separately so that the proportion of PVE by the two groups are 0.2 and 0.8, respectively. Again, we set the total PVE to be either 0.2, 0.5, or 0.8. This simulation scenario consists of three simulation settings for each PVE ($c = 10$, 100, or 1000).

Scenario IV is related to the assumption made in LMM and MultiBLUP. Here, all SNPs have non-zero effects and their effect sizes come from either a normal distribution, a t-distribution with four degrees of freedom, or a Laplace distribution. We scaled their effect sizes further so that the total PVE equals 0.2, 0.5, or 0.8. This simulation scenario consists of three simulation settings for each PVE (normal, t, or Laplace).

In each setting, we performed 20 simulation replicates. In each replicate, we randomly split the data into a training data with 80% individuals and a test data with the remaining 20% individuals. We then fitted different methods on the training data and evaluated their prediction performance on the test data (i.e., Monte Carlo cross validation).

GEUVADIS data. The GEUVADIS data[42] contains gene expression measurements for 465 individuals from five different populations: CEPH (CEU), Finns (FIN), British (GBR), Toscani (TSI), and Yoruba (YRI). Following previous studies[65], we focused only on protein coding genes and lincRNAs that are annotated from GENCODE[66] (release 12). We removed lowly expressed genes that have zero counts in at least half of the individuals and obtained a final set of 15,810 genes. Afterwards, following previous studies[67], we performed PEER normalization to remove confounding effects and unwanted variations. In order to remove potential population stratification, we quantile normalized the gene expression measurements across individuals in each population to a standard normal distribution, and then quantile normalized the gene expression measurements to a standard normal distribution across individuals from all five populations. In addition to the gene expression data, all individuals in GEUVADIS also have their genotypes sequenced in the 1000 Genomes Project. Among the sequenced genotypes, we filtered out SNPs that have a Hardy-Weinberg equilibrium (HWE) p-value $< 10^{-4}$, a genotype call rate < 95%, or an MAF < 0.01. We retained a total of 7,072,917 SNPs for analysis. We intersected these SNPs with imputed SNPs from WTCCC data[4] (see below; for the purpose of performing gene set tests) and kept a final set of 2,793,818 overlapping SNPs for analysis. Then, for each gene in turn, we obtained its cis-SNPs that are located within either 100 kb upstream of the transcription start site or 100 kb downstream of the transcription end site, resulting in an average of 175 cis-SNPs per gene.

WTCCC data. The WTCCC[4] 1 data consists of about 14,000 cases from seven common diseases and 2938 shared controls. The cases include 1963 individuals with type 1 diabetes (T1D), 1748 individuals with Crohn's disease (CD), 1860 individuals with rheumatoid arthritis (RA), 1868 individuals with bipolar disorder (BD), 1924 individuals with type 2 diabetes (T2D), 1926 individuals with coronary artery disease (CAD), and 1952 individuals with hypertension (HT). We obtained quality controlled genotypes from WTCCC and imputed missing genotypes using BIMBAM[68]. We obtained a total of 458,868 SNPs shared across all individuals. We then further imputed SNPs using the 1000 Genomes as the reference panel with SHAPEIT and IMPUTE2[69]. We filtered out SNPs that have a HWE p-value $< 10^{-4}$, a genotype call rate < 95%, or an MAF < 0.01 to obtain a total of 2,793,818 imputed SNPs. For PrediXcan analysis[12], as in the GEUVADIS data (see above), we focused on the same 15,810 genes. As in ref. [12], we further restricted our association analysis on a set of 4343 genes that have a predictive R^2 above 0.01 by all predictive methods.

Cattle data. The cattle data[17] consists of 5024 samples and 42,551 SNPs after removing SNPs that have a HWE p-value $< 10^{-4}$, a genotype call rate <95%, or an MAF < 0.01. For the remaining SNPs, we imputed missing genotypes with the estimated mean genotype of that SNP. We analyzed three traits: MFP, MY, and SCS. All phenotypes were quantile normalized to a standard normal distribution before analysis.

Maize data. The maize data[15] consists of 2267 inbred accessions and 98,385 SNPs after removing SNPs that have a HWE p-value $< 10^{-4}$, a genotype call rate <95%, or an MAF < 0.01. For the remaining SNPs, we imputed missing genotypes with the estimated mean genotype of that SNP. We used the GDDs to silking as the phenotype in genomic selection. GDD was calculated using climate data from weather stations located near the farms[15], and was quantile normalized to a standard normal distribution before analysis.

FHS data. The FHS data contains genotype data on 6950 individuals and 394,174 SNPs. We filtered out SNPs that have a HWE p-value $< 10^{-4}$, a genotype call rate <95%, or an MAF < 0.01 to obtain a final set of 387,741 SNPs. For these SNPs, we imputed missing genotypes with the estimated mean genotype of that SNP. We performed analysis on eight traits: five commonly used plasma traits that include LDL cholesterol, GLU, HDL cholesterol, TC, and TGs; and three anthropometric traits that include height, weight, and BMI. Each trait was quantile normalized to a standard normal distribution before analysis. Note that the FHS data is a family-based study where individuals are genetically related. To tease apart the influence of individual relatedness on prediction performance among methods, we also divided the samples in FHS into two separate data sets with different levels of relatedness. Specifically, we first used genotypes to compute the genome-wide genetic relatedness matrix. We then ordered individual pairs based on their genetic relatedness values. From top to bottom of the ordered individual pair list, we selected individuals from individual pairs with high levels of relatedness into a data set D1, and continued this process until the sample size of D1 was half of the full data. We then kept the remaining individuals from individual pairs with low levels of relatedness into a data set D2. The relatedness threshold for separating individual pairs between the two data sets was 0.151. Nevertheless, the majority pairs in D1 and D2 have genetic relatedness values close to zero: 99.6% of pairs in D1 and 99.9% of pairs in D2 have a genetic relatedness value between +/−0.01. As another way of measuring relatedness, we also computed the effective number of chromosome segments (M_e)[49] in the two data. M_e is a crucial parameter that measures the effective number of independent SNPs and is also closely related to the effective number of independent individuals: M_e is small in a data with related individuals and is large in a data with unrelated individuals. A small value of M_e often correlates to high prediction accuracy[48, 49, 70]. With 20 cross-validation replicates, we estimated M_e in D1 and D2 sub data to be 34541.39 (sd = 140.87) and 81786.01 (sd = 651.52), respectively.

Other methods. We compared the performance of DPR.MCMC and DPR.VB mainly with five existing methods: (1) LMM[28] as implemented in the GEMMA software (version 0.95alpha); (2) BVSR[29] as implemented in the GEMMA software (version 0.95alpha); (3) MultiBLUP[41] as implemented in the LDAK software (version 4.9); (4) BayesR[35] as implemented in the bayesR software; (5) rjMCMC[20] as implemented in the gwas_rjmc1.163 software. We used default settings to fit all these methods. For rjMCMC, because it requires us to provide a variance component parameter, we used LMM to estimate the variance component first in all analyses. In addition, rjMCMC does not output parameter estimates. Therefore, for the PrediXcan analysis, we first merged the GEUVADIS and WTCCC files for every gene, labeled WTCCC individuals as having missing phenotypes, and then ran rjMCMC on these files to obtained predicted gene expression values using the WTCCC genotype data. The same strategy was also applied to perform cross-validation prediction between D1 and D2 sub data sets in FHS. For gene expression prediction and PrediXcan analysis, following the original PrediXcan paper[12], we also used ENET[44], which is implemented in the R package glmnet (version 1.9-5). For ENET, following[12], we set one penalty parameter (i.e., α) to be 0.5 and selected the other one using 100-fold cross validation in the training data.

Code availability. Our method is implemented in the DPR software, freely available at http://www.xzlab.org/software.html.

Data availability. No data were generated in the present study. The GEUVADIS gene expression data is publicly available at http://www.geuvadis.org. The genotype data from the 1000 genomes project is publicly available at http://www.internationalgenome.org. The WTCCC genotype and phenotype data is publicly available at https://www.wtccc.org.uk. The genotype and phenotype data from the cattle and maize studies are available from the authors upon reasonable request and with permission of Prof. Xiaolei Liu at the HuaZhong Agriculture University (xiaoleiliu@mail.hzau.edu.cn). The FHS genotype and phenotype data is available in dbGaP (https://www.ncbi.nlm.nih.gov/gap) with accession number phs000007.

Received: 26 November 2016 Accepted: 30 June 2017
Published online: 06 September 2017

References

1. Fritsche, L. G. et al. A large genome-wide association study of age-related macular degeneration highlights contributions of rare and common variants. *Nat. Genet.* **48**, 134–143 (2016).

2. Mancuso, N. et al. The contribution of rare variation to prostate cancer heritability. *Nat. Genet.* **48**, 30–35 (2016).
3. Fuchsberger, C. et al. The genetic architecture of type 2 diabetes. *Nature* **536**, 41–47 (2016).
4. The Wellcome Trust Case Control Consortium. Genome-wide association study of 14,000 cases of seven common diseases and 3000 shared controls. *Nature* **447**, 661–678 (2007).
5. Global Lipids Genetics Consortium. Discovery and refinement of loci associated with lipid levels. *Nat. Genet.* **45**, 1274–1283 (2013).
6. Afshari, N. A. et al. Genome-wide association study identifies three novel loci in Fuchs endothelial corneal dystrophy. *Nat. Commun.* **8**, 14898 (2017).
7. Hoffmann, T. J. et al. Genome-wide association study of prostate-specific antigen levels identifies novel loci independent of prostate cancer. *Nat. Commun.* **8**, 14248 (2017).
8. Warren, H. R. et al. Genome-wide association analysis identifies novel blood pressure loci and offers biological insights into cardiovascular risk. *Nat. Genet.* **49**, 403–415 (2017).
9. Makowsky, R. et al. Beyond missing heritability: Prediction of complex traits. *PLoS Genet.* **7**, e1002051 (2011).
10. Hayes, B. J., Pryce, J., Chamberlain, A. J., Bowman, P. J. & Goddard, M. E. Genetic architecture of complex traits and accuracy of genomic prediction: Coat colour, milk-fat percentage, and type in holstein cattle as contrasting model traits. *PLoS Genet.* **6**, e1001139 (2010).
11. Chatterjee, N., Shi, J. & Garcia-Closas, M. Developing and evaluating polygenic risk prediction models for stratified disease prevention. *Nat. Rev. Genet.* **17**, 392–406 (2016).
12. Gamazon, E. R. et al. A gene-based association method for mapping traits using reference transcriptome data. *Nat. Genet.* **47**, 1091–1098 (2015).
13. Allen, H. L. et al. Hundreds of variants clustered in genomic loci and biological pathways affect human height. *Nature* **467**, 832–838 (2010).
14. Li, H. et al. Genome-wide association study dissects the genetic architecture of oil biosynthesis in maize kernels. *Nat. Genet.* **45**, 43–50 (2013).
15. Romay, M. C. et al. Comprehensive genotyping of the USA national maize inbred seed bank. *Genome Biol.* **14**, R55 (2013).
16. Fernandes Júnior, G. A. et al. Genomic prediction of breeding values for carcass traits in Nellore cattle. *Genet. Sel. Evol.* **48**, 7 (2016).
17. Zhang, Z. et al. Accuracy of whole-genome prediction using a genetic architecture-enhanced variance-covariance matrix. *G3* **5**, 615–627 (2015).
18. Meuwissen, T., Hayes, B. & Goddard, M. Prediction of total genetic value using genome-wide dense marker maps. *Genetics* **157**, 1819–1829 (2001).
19. de los Campos, G., Vazquez, A. I., Fernando, R., Klimentidis, Y. C. & Sorensen, D. Prediction of complex human traits using the genomic best linear unbiased predictor. *PLoS Genet.* **9**, e1003608 (2013).
20. Lee, S. H., van der Werf, J. H. J., Hayes, B. J., Goddard, M. E. & Visscher, P. M. Predicting unobserved phenotypes for complex traits from whole-genome SNP data. *PLoS Genet.* **4**, e1000231 (2008).
21. Hayes, B., Bowman, P., Chamberlain, A. & Goddard, M. Genomic selection in dairy cattle: Progress and challenges. *J. Dairy Sci.* **92**, 433–443 (2009).
22. Goddard, M. E. & Hayes, B. Genomic selection. *J. Anim. Breed. Genet.* **124**, 323–330 (2007).
23. Meuwissen, T., Hayes, B. & Goddard, M. Accelerating improvement of livestock with genomic selection. *Annu. Rev. Anim. Biosci* **1**, 221–237 (2013).
24. Chatterjee, N. et al. Projecting the performance of risk prediction based on polygenic analyses of genome-wide association studies. *Nat. Genet.* **45**, 400–405 (2013).
25. Shah, S. et al. Improving phenotypic prediction by combining genetic and epigenetic associations. *Am. J. Hum. Genet.* **97**, 75–85 (2015).
26. Maier, R. et al. Joint analysis of psychiatric disorders increases accuracy of risk prediction for schizophrenia, bipolar disorder, and major depressive disorder. *Am. J. Hum. Genet.* **96**, 283–294 (2015).
27. Weissbrod, O., Geiger, D. & Rosset, S. Multikernel: Linear mixed models for complex phenotype prediction. *Genome Res.* **26**, 969–979 (2016).
28. Yang, J. et al. Common SNPs explain a large proportion of the heritability for human height. *Nat. Genet.* **42**, 565–569 (2010).
29. Habier, D., Fernando, R. L., Kizilkaya, K. & Garrick, D. J. Extension of the bayesian alphabet for genomic selection. *BMC Bioinformatics* **12**, 186 (2011).
30. Park, T. & Casella, G. The bayesian lasso. *J. Am. Stat. Assoc.* **103**, 681–686 (2008).
31. Yi, N. & Xu, S. Bayesian LASSO for quantitative trait loci mapping. *Genetics* **179**, 1045–1055 (2008).
32. Hoggart, C. J., Whittaker, J. C., De Iorio, M. & Balding, D. J. Simultaneous analysis of all SNPs in genome-wide and re-sequencing association studies. *PLoS Genet.* **4**, e1000130 (2008).
33. Guan, Y. & Stephens, M. Bayesian variable selection regression for genome-wide association studies and other large-scale problems. *Ann. Appl. Stat* **5**, 1780–1815 (2011).
34. Zhou, X., Carbonetto, P. & Stephens, M. Polygenic modeling with Bayesian sparse linear mixed models. *PLoS Genet.* **9**, e1003264 (2013).
35. Moser, G. et al. Simultaneous discovery, estimation and prediction analysis of complex traits using a bayesian mixture model. *PLoS Genet.* **11**, e1004969 (2015).
36. Goddard, M. Genomic selection: prediction of accuracy and maximisation of long term response. *Genetica* **136**, 245–257 (2009).
37. Ghahramani, Z. Bayesian non-parametrics and the probabilistic approach to modelling. *Philos. T. R. Soc. A* **371**, 20110553 (2013).
38. Müller, P. & Mitra, R. Bayesian nonparametric inference—why and how. *Bayesian Anal.* **8**, 269–302 (2013).
39. Gershman, S. J. & Blei, D. M. A tutorial on Bayesian nonparametric models. *J. Math. Psychol.* **56**, 1–12 (2012).
40. Müller, P. & Quintana, F. A. Nonparametric bayesian data analysis. *Stat. Sci.* **19**, 95–110 (2004).
41. Speed, D. & Balding, D. J. MultiBLUP: improved SNP-based prediction for complex traits. *Genome Res.* **24**, 1550–1557 (2014).
42. Lappalainen, T. et al. Transcriptome and genome sequencing uncovers functional variation in humans. *Nature* **501**, 506–511 (2013).
43. 1000 Genomes Project Consortium. An integrated map of genetic variation from 1092 human genomes. *Nature* **491**, 56–65 (2012).
44. Zou, H. & Hastie, T. Regularization and variable selection via the Elastic Net. *J. R. Stat. Soc. Ser. B* **67**, 301–320 (2005).
45. Splansky, G. L. et al. The third generation cohort of the national heart, lung, and blood institute's framingham heart study: design, recruitment, and initial examination. *Am. J. Epidemiol.* **165**, 1328–1335 (2007).
46. Hu, Z. L., Park, C. A., Wu, X. L. & Reecy, J. M. Animal QTLdb: an improved database tool for livestock animal QTL/association data dissemination in the post-genome era. *Nucleic Acids Res.* **41**, D871–D879 (2013).
47. Spiliopoulou, A. et al. Genomic prediction of complex human traits: relatedness, trait architecture and predictive meta-models. *Hum. Mol. Genet.* **24**, 4167–4182 (2015).
48. Goddard, M. E., Hayes, B. J. & Meuwissen, T. H. E. Using the genomic relationship matrix to predict the accuracy of genomic selection. *J. Anim. Breed. Genet.* **128**, 409–421 (2011).
49. Lee, S. H., Weerasinghe, W. M. S. P., Wray, N. R., Goddard, M. E. & van der Werf, J. H. J. Using information of relatives in genomic prediction to apply effective stratified medicine. *Sci. Rep.* **7**, 42091 (2017).
50. Carbonetto, P. & Stephens, M. Scalable variational inference for Bayesian variable selection in regression, and its accuracy in genetic association studies. *Bayesian Anal.* **7**, 73–108 (2012).
51. Yi, H., Breheny, P., Imam, N., Liu, Y. & Hoeschele, I. Penalized multimarker vs. single-marker regression methods for genome-wide association studies of quantitative traits. *Genetics* **199**, 205–222 (2015).
52. Sun, S. *et al.* Differential expression analysis for RNAseq using Poisson mixed models. *Nucleic Acids Res.* gkx204. doi: 210.1093/nar/gkx1204 (2017).
53. Tung, J., Zhou, X., Alberts, S. C., Stephens, M. & Gilad, Y. The genetic architecture of gene expression levels in wild baboons. *Elife* **4**, e04729 (2015).
54. Zhou, X. et al. Epigenetic modifications are associated with inter-species gene expression variation in primates. *Genome Biol.* **15**, 1 (2014).
55. Lea, A. J., Tung, J. & Zhou, X. A Flexible, efficient binomial mixed model for identifying differential DNA methylation in bisulfite sequencing data. *PLoS Genet.* **11**, e1005650 (2015).
56. Manolio, T. et al. Finding the missing heritability of complex diseases. *Nature* **461**, 747–753 (2009).
57. Shi, J. et al. Winner's curse correction and variable thresholding improve performance of polygenic risk modeling based on genome-wide association study summary-level data. *PLoS Genet.* **12**, e1006493 (2016).
58. Li, J., Das, K., Fu, G., Li, R. & Wu, R. The Bayesian lasso for genome-wide association studies. *Bioinformatics* **27**, 516–523 (2011).
59. Blei, D. M. & Jordan, M. I. Variational inference for Dirichlet process mixtures. *Bayesian Anal.* **1**, 121–143 (2006).
60. Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. Variational inference: a review for statisticians. *J. Am. Stat. Assoc.* **112**, 859–877 (2017).
61. Ranganath, R., Tran, D. & Blei, D. M. *Hierarchical Variational Models.* Paper presented at the International Conference on Machine Learning (2016).
62. Zhou, X. A Unified framework for variance component estimation with summary statistics in genome-wide association studies. *Ann. Appl. Stat.* (in press), Preprint at http://biorxiv.org/content/early/2016/03/08/042846 (2017).
63. Andrews, D. F. & Mallows, C. L. Scale mixtures of normal distributions. *J. R. Stat. Soc. Ser. B* **36**, 99–102 (1974).
64. Verbyla, K. L., Hayes, B. J., Bowman, P. J. & Goddard, M. E. Accuracy of genomic selection using stochastic search variable selection in Australian Holstein Friesian dairy cattle. *Genet. Res.* **91**, 307–311 (2009).
65. Wen, X., Luca, F. & Pique-Regi, R. Cross-population joint analysis of eQTLs: Fine mapping and functional annotation. *PLoS Genet.* **11**, e1005176 (2015).

66. Harrow, J. et al. GENCODE: the reference human genome annotation for The ENCODE Project. *Genome Res.* **22**, 1760–1774 (2012).
67. Stegle, O., Parts, L., Piipari, M., Winn, J. & Durbin, R. Using probabilistic estimation of expression residuals (PEER) to obtain increased power and interpretability of gene expression analyses. *Nat. Protoc.* **7**, 500–507 (2012).
68. Guan, Y. & Stephens, M. Practical issues in imputation-based association mapping. *PLoS Genet.* **4**, e1000279 (2008).
69. Howie, B. N., Donnelly, P. & Marchini, J. A flexible and accurate genotype imputation method for the next generation of genome-wide association studies. *PLoS Genet.* **5**, e1000529 (2009).
70. Lee, S. H., Clark, S. & van der Werf, J. Estimation of genomic prediction accuracy from reference populations with varying degrees of relationship. bioRxiv, Preprint at http://biorxiv.org/content/early/2017/03/22/119164 (2017).

Acknowledgements

This research is supported by National Institutes of Health grant R01HG009124. We thank Xiaolei Liu from the HuaZhong Agriculture University (xiaoleiliu@mail.hzau.edu.cn) for providing us with the cattle and maize data. We thank Dr Shiquan Sun for help with the development of DPR. We thank Dr Doug Speed for help with LDAK and Dr Sang Hong Lee for help with gwas_rjmc1.163 software. This study also makes use of data generated by the Wellcome Trust Case Control Consortium (WTCCC). A full list of the investigators who contributed to the generation of the data is available from http://www.wtccc.org.uk. Funding for the WTCCC project was provided by the Wellcome Trust under award 076113 and 085475. This research was conducted in part using data and resources from the Framingham Heart Study of the NHLBI and Boston University School of Medicine, which was partially supported by the NHLBI Framingham Heart Study (Contract No. N01-HC-25195) and its contract with Affymetrix, Inc for genotyping services (Contract No. N02-HL-6-4278). We thank all participants and staff from the Framingham Heart Study.

Author contributions

X.Z. and P.Z. conceived and designed the experiments, developed the algorithm, and implemented the software used in analysis. P.Z. performed the experiments, analyzed the data. X.Z. and P.Z. wrote the paper.

Additional information

Supplementary Information accompanies this paper at doi:10.1038/s41467-017-00470-2.

Competing interests: The authors declare no competing financial interests.

Reprints and permission information is available online at http://npg.nature.com/reprintsandpermissions/

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

File name: Supplementary Information
Description: Supplementary figures, supplementary tables, supplementary notes and supplementary references.

Supplementary Note: Model and Algorithm Details for DPR

The latent Dirichlet process regression model

We consider the following multiple linear regression model

$$\mathbf{y} = \mathbf{W}\boldsymbol{\alpha} + \mathbf{X}\tilde{\boldsymbol{\beta}} + \boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim N(0, \sigma_e^2 \mathbf{I}_n), \quad (1)$$

where $\mathbf{y}$ is an n-vector of phenotypes measured on n individuals; $\mathbf{W}$ is an n by c matrix of covariates including a column of 1s for the intercept term; $\boldsymbol{\alpha}$ is a c-vector of coefficients; $\mathbf{X}$ is an n by p matrix of genotypes; $\tilde{\boldsymbol{\beta}}$ is the corresponding p-vector of effect sizes; $\boldsymbol{\varepsilon}$ is an n-vector of residual errors where each element is assumed to be independently and identically distributed from a normal distribution with variance σ_e^2. Note that we use $\tilde{\boldsymbol{\beta}}$ here instead of $\boldsymbol{\beta}$ as in the main text for reasons that will become clear shortly.

As explained in the main text, we assign a normal prior $N(0, \sigma^2\sigma_e^2)$ on each element of $\tilde{\boldsymbol{\beta}}$, and we further assign a Dirichlet process prior on the variance parameter σ^2. (Note that different from the main text, we also scale the variance with the error variance σ_e^2 to simply the algorithm.) Integrating out σ^2 induces a Dirichlet process normal mixture prior on $\tilde{\beta}_i$

$$\begin{aligned} \tilde{\beta}_i &\sim \sum_{k=1}^{\infty} \pi_k N(0, (\sigma_k^2 + \sigma_b^2)\sigma_e^2), \\ \pi_k &= \nu_k \prod_{l=1}^{k-1} (1 - \nu_l), \nu_k \sim \text{Beta}(1, \lambda), \end{aligned} \quad (2)$$

where $\sigma_k^2 + \sigma_b^2$ (scaled by σ_e^2) is the variance for each normal component. Again, to simply the algorithm, different from the main text, we add a common variance σ_b^2 to each variance component and we set $\sigma_k^2 = 0$ when $k = 1$. We refer to the above model based on equations (1) and (2) as the latent Dirichlet Process Regression (DPR) model. For the hyper-parameters $\boldsymbol{\alpha}$, σ_k^2, σ_b^2, σ_e^2, and λ in the model, we consider the following priors

$$
\begin{aligned}
\alpha_j &\sim N(0, \sigma_e^2\sigma_w^2), \sigma_w^2 \to \infty, \\
\sigma_k^2 &\sim \text{inverse-gamma } (a_{0k}, b_{0k}), \\
\sigma_b^2 &\sim \text{inverse-gamma } (a_{0b}, b_{0b}), \\
\sigma_e^2 &\sim \text{inverse-gamma } (a_{0e}, b_{0e}), \\
\lambda &\sim \text{gamma}(a_{0\lambda}, b_{0\lambda}),
\end{aligned} \tag{3}
$$

where we set $a_{0k}, b_{0k}, a_{0b}, b_{0b}, a_{0e}$, and b_{0e} in the inverse gamma distributions to be 0.1; we set $a_{0\lambda}$ and $b_{0\lambda}$ in the gamma distribution to be 1 and 0.1; and we use a limiting normal prior for each α_j with the normal variance goes to infinity, since generally there is enough information in the likelihood to overwhelm any reasonable prior assumption for these parameters.

To improve mixing, following[1], we group the effect sizes that correspond to the first normal component with the smallest variance σ_b^2 in equation (2) into a random effects term $\mathbf{u}$:

$$
\mathbf{u} = \mathbf{Xb} \sim N(0, \sigma_b^2\sigma_e^2\mathbf{K}), \tag{4}
$$

where $\mathbf{K} = \mathbf{XX}^T / p$ is the genetic relatedness matrix (GRM)[1,2] computed using centered SNPs. Note that the GRM is typically positive semi-definite with one eigen-value being zero due to genotype centering. We do not need to deal with the zero eigenvalue because our algorithms do not involve the inverse of GRM. This way, the model in equation (1) becomes

$$
\mathbf{y} = \mathbf{W\alpha} + \mathbf{X\beta} + \mathbf{u} + \boldsymbol{\varepsilon}, \boldsymbol{\varepsilon} \sim N(0, \sigma_e^2\mathbf{I}_n), \tag{5}
$$

explaining our use of $\tilde{\boldsymbol{\beta}}$ in equation (1). In our notation, $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta} + \mathbf{b}$. The corresponding prior on each element of $\mathbf{b}$ is

$$
b_i \sim N(0, \sigma_b^2\sigma_e^2 / p), \tag{6}
$$

and the corresponding prior on each element of $\boldsymbol{\beta}$ is

$$
\beta_i \sim \pi_1 N(0, 0 \times \sigma_e^2) + \sum_{k=2}^{\infty} \pi_k N(0, \sigma_k^2\sigma_e^2). \tag{7}
$$

We will develop algorithms for fitting the equivalent model defined in equation (5) in the following text. With the fitting algorithm, we can obtain the posterior mean of $\tilde{\boldsymbol{\beta}}$ as the

sum of the posterior mean of $\boldsymbol{\beta}$ and the posterior mean of $\mathbf{b}$. We use the posterior mean of $\tilde{\boldsymbol{\beta}}$ to compute prediction errors.

Difference between DPR and BayesR

Before we proceed further, it is useful to clarify the difference between DPR and the previously proposed method BayesR[3]. While our method is motivated in part by BayesR, DPR is different from BayesR in five important areas. First, BayesR is a sparse model while DPR is a non-sparse model: BayesR assumes that most SNPs have zero effects while DPR assumes that all SNPs have non-zero effects. As a result, BayesR and DPR are expected to perform differently in sparse vs non-sparse settings. Second, BayesR fixes the ratio between the variance parameters from the three non-zero components to be 0.01:0.1:1. In contrast, DPR estimates the variance of all non-zero components from the data at hand. Inferring parameters from the data instead of fixing them to pre-set values is expected to improve prediction performance. Third, BayesR uses a mixture of three normal distributions for the non-zero component, while DPR uses infinitely many normal distributions *a priori*. Using three normals can sometimes fail to capture the complicated effect size distributions encountered in a range of genetic architectures, as is evident in simulations presented in the main text. Fourth, importantly, it is not straightforward to extend BayesR to accommodate a larger number of normal components. Consequently, while the BayesR software allows users to specify an arbitrary number of components, in those analyses, BayesR also requires users to provide the variance component estimates for these components. It is far from trivial to figure out how one should obtain these variance component estimates for BayesR. In contrast, DPR provides a principled way to extend the simple normal model to accommodate a much larger number of normal components, ensuring robust prediction performance across a range of settings. Fifth, as we will show below, we fix the number of normal components in DPR in practice due to computational reasons. As has been previously shown in other settings[11,12], using a small number of components to approximate the Dirichlet process can undermine its performance. Therefore, we do want to acknowledge that the results we present in the main text are likely conservative estimates of DPR's performance. Better approximations to the Dirichlet process may improve DPR's prediction performance further.

MCMC sampling

Here, we describe our Markov Chain Monte Carlo (MCMC) sampling algorithm to obtain the posterior samples from DPR. To facilitate MCMC, for each SNP i, we assign a vector of indicator variables $\gamma_{ik} \in \{0, 1\}$ to indicate which normal component β_i comes from. To improve convergence, we integrate out $\mathbf{u}$ in model (5) and then perform Gibbs sampling by using the conditional distributions for each parameter in turn. Specifically, let $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma_b^2, \sigma_k^2, \nu_k, \gamma_{ik}, \lambda, \sigma_e^2)$ includes all unknown parameters in model (5), our joint log marginal posterior after integrating out $\mathbf{u}$ is

$$
\begin{aligned}
\log p(\boldsymbol{\theta} \mid \mathbf{y}) &= \log p(\mathbf{y} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma_b^2, \sigma_e^2) + \log p(\boldsymbol{\beta} \mid \boldsymbol{\gamma}, \sigma_k^2, \sigma_e^2) \\
&\quad + \log p(\boldsymbol{\gamma} \mid \nu_k) + \log p(\nu_k \mid \lambda) + \log p(\sigma_k^2 \mid a_{0k}, b_{0k}) + \log p(\sigma_e^2 \mid a_{0e}, b_{0e}) \\
&\quad + \log p(\sigma_b^2 \mid a_{0b}, b_{0b}) + \log p(\lambda \mid a_{0\lambda}, b_{0\lambda}) \\
&= C - \frac{1}{2}\log \mid \sigma_e^2 \mathbf{H} \mid - \frac{1}{2\sigma_e^2}(\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta})^T \mathbf{H}^{-1}(\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta}) \\
&\quad + \sum_i \sum_{k=2}^{\infty} \gamma_{ik}\left(-\frac{1}{2}\log(\sigma_e^2) - \frac{1}{2}\log(\sigma_k^2) - \frac{\beta_{ik}^2}{2\sigma_k^2\sigma_e^2}\right) \\
&\quad + \sum_i \sum_k^{\infty} \gamma_{ik}\left(\log(\nu_k) + \sum_{l=1}^{k-1}\log(1-\nu_l)\right) + \sum_k^{\infty}\left((\lambda - 1)\log(1-\nu_k) + \log(\lambda)\right) \\
&\quad - \sum_k^{\infty}(a_{0k} + 1)\log(\sigma_k^2) - \sum_k^{\infty} b_{0k}\sigma_k^{-2} - (a_{0e} + 1)\log(\sigma_e^2) - b_{0e}\sigma_e^{-2} \\
&\quad - (a_{0b} + 1)\log(\sigma_b^2) - b_{0b}\sigma_b^{-2} + (a_{0\lambda} - 1)\log(\lambda) - b_{0\lambda}\lambda,
\end{aligned}
\tag{8}
$$

where $\mathbf{H} = \mathbf{I}_n + \sigma_b^2 \mathbf{K}$ and C is a normalizing constant. To simplify notation, we will ignore all constant terms from now on. Based on the joint posterior, we can derive the conditional posterior distribution for each parameter in turn. When we derive these conditional distributions, we will also ignore the other parameters which these distributions are conditional on to simplify the presentation.

Sampling α_j

First, for α_j we have

$$
\log p(\alpha_j \mid .) = -\frac{\sigma_e^{-2}\mathbf{w}_j{}^T\mathbf{H}^{-1}\mathbf{w}_j}{2}\alpha_j^2 + \sigma_e^{-2}\mathbf{w}_j{}^T\mathbf{H}^{-1}(\mathbf{y} - \sum_{m \neq j}\mathbf{w}_m\alpha_m - \mathbf{X}\boldsymbol{\beta})\alpha_j. \tag{9}
$$

Therefore, the conditional distribution for sampling α_j is $p(\alpha_j \mid .) = N(m_j, s_j^2)$, where

$$m_j = \frac{\mathbf{w}_j{}^T \mathbf{H}^{-1}(\mathbf{y} - \sum_{m \neq j} \mathbf{w}_m \alpha_m - \mathbf{X}\boldsymbol{\beta})}{\mathbf{w}_j{}^T \mathbf{H}^{-1} \mathbf{w}_j},$$
$$s_j^2 = \frac{\sigma_e^2}{\mathbf{w}_j{}^T \mathbf{H}^{-1} \mathbf{w}_j}. \quad (10)$$

Sampling β_{ik} and γ_{ik}

For β_{ik} and γ_{ik}, we have

$$\log p(\beta_{ik}, \gamma_{ik} \mid .) = -\frac{\sigma_e^{-2} \mathbf{x}_i^T \mathbf{H}^{-1} \mathbf{x}_i}{2} \beta_i^2 + \sigma_e^{-2} \mathbf{x}_i^T \mathbf{H}^{-1} (\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \sum_{m \neq i} \mathbf{x}_m \beta_m) \beta_i$$
$$+ \gamma_{ik} (-\frac{1}{2} \log(\sigma_e^2) - \frac{1}{2} \log(\sigma_k^2) - \frac{1}{2} \sigma_e^{-2} \sigma_k^{-2} \beta_{ik}^2) + \gamma_{ik} (\log(\nu_k) + \sum_{l=1}^{k-1} \log(1 - \nu_l)). \quad (11)$$

Therefore, the conditional distributions for sampling β_{ik} and γ_{ik} are

$$p(\beta_{ik} \mid \gamma_{ik} = 1, .) = N(m_{ik}, s_{ik}^2),$$
$$p(\gamma_{ik} = 1 \mid .) = \pi_{ik} \propto e^{m_{ik}^2 / 2 s_{ik}^2 + \log(s_{ik}) - \log(\sigma_e) - \log(\sigma_k) + \log(\nu_k) + \sum_{l=1}^{k-1} \log(1 - \nu_l)}, \quad (12)$$

where

$$m_{ik} = \frac{\mathbf{x}_i^T \mathbf{H}^{-1} (\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \sum_{m \neq i} \mathbf{x}_m \beta_m)}{\mathbf{x}_i^T \mathbf{H}^{-1} \mathbf{x}_i + \sigma_k^{-2}},$$
$$s_{ik}^2 = \frac{\sigma_e^2}{\mathbf{x}_i^T \mathbf{H}^{-1} \mathbf{x}_i + \sigma_k^{-2}}. \quad (13)$$

Sampling v_k

For v_k, we have

$$\log p(\nu_k \mid .) = \sum_i \gamma_{ik} \log(\nu_k) + \sum_i \sum_{l=k+1}^{\infty} \gamma_{il} \log(1 - \nu_k) + (\lambda - 1) \log(1 - \nu_k). \quad (14)$$

Therefore, the conditional distribution for sampling v_k is $p(\nu_k \mid .) = \text{Beta}(\kappa_k, \lambda_k)$, where

$$\kappa_k = \sum_i \gamma_{ik} + 1,$$
$$\lambda_k = \sum_i \sum_{l=k+1}^{\infty} \gamma_{il} + \lambda. \quad (15)$$

Sampling σ_k^2

For σ_k^2, we have

$$\log p(\sigma_k^2 | .) = -(\frac{\sum_i \gamma_{ik}}{2} + a_{0k} + 1)\log(\sigma_k^2) - (\frac{\sum_i \gamma_{ik}\beta_{ik}^2\sigma_e^{-2}}{2} + b_{0k})\sigma_k^{-2}. \quad (16)$$

Therefore, the conditional distribution for sampling σ_k^2 is $p(\sigma_k^2 | .) = \text{inverse-gamma}(a_k, b_k)$, where

$$\begin{aligned} a_k &= \frac{1}{2}\sum_i \gamma_{ik} + a_{0k}, \\ b_k &= \frac{1}{2\sigma_e^2}\sum_i \gamma_{ik}\beta_{ik}^2 + b_{0k}. \end{aligned} \quad (17)$$

Sampling λ

For λ, we have

$$\log p(\lambda|.) = \lambda(\sum_k^{\infty}\log(1-\nu_k) - b_{0\lambda}) + \log(\lambda)(a_{0\lambda} + \sum_k^{\infty} 1_k). \quad (18)$$

Therefore, the conditional distribution for sampling λ is $p(\lambda|.) = \text{gamma}(a_\lambda, b_\lambda)$, where

$$\begin{aligned} a_\lambda &= a_{0\lambda} + \sum_k^{\infty} 1_k, \\ b_\lambda &= b_{0\lambda} - \sum_k^{\infty}\log(1-\nu_k). \end{aligned} \quad (19)$$

Sampling σ_e^2

For σ_e^2, we have

$$\begin{aligned} \log p(\sigma_e^2 | .) = &-((n + \sum_i\sum_{k=2}\gamma_{ik})/2 + a_{0e} + 1)\log(\sigma_e^2) - \frac{1}{2}\text{SSR}\times\sigma_e^{-2} \\ &-\frac{1}{2}(\sum_i\sum_k \gamma_{ik}\beta_{ik}^2\sigma_k^{-2} + 2b_{0e})\sigma_e^{-2}. \end{aligned} \quad (20)$$

Therefore, the conditional distribution for sampling σ_e^2 is $p(\sigma_e^2 | .)$ = inverse-gamma(a_e, b_e) where

$$a_e = n/2 + \sum_i \sum_{k=2} \gamma_{ik}/2 + a_{0e},$$

$$b_e = \frac{1}{2}(\mathrm{SSR} + \sum_i \sum_{k=2} \gamma_{ik}\beta_{ik}^2/\sigma_k^2 + 2b_{0e}), \quad (21)$$

$$\mathrm{SSR} = (\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta})^T \mathbf{H}^{-1}(\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta}).$$

Sampling σ_b^2

For σ_b^2, we have

$$\log p(\sigma_b^2|.) = -\frac{1}{2}\log|\mathbf{H}| - \frac{1}{2\sigma_e^2}(\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta})^T \mathbf{H}^{-1}(\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta}) - (a_{0b}+1)\log(\sigma_b^2) - b_{0b}\sigma_b^{-2}, \quad (22)$$

which is in an unknown distributional form. Nevertheless, it is straightforward to sample from this univariate distribution using reject sampling, importance sampling or other standard methods[4]. Here, we sample σ_b^2 based on re-parameterization of σ_b^2 following[1,5]. Specifically, we define a new parameter (h^2)[2,6,7]

$$h^2 = \frac{\sigma_b^2}{\sigma_b^2 + 1}, \quad (23)$$

which is bounded between 0 and 1. The log-posterior conditional distribution for h^2 is

$$\log p(h^2|.) = \log p(\sigma_b^2(h^2)|.) - 2\log(1-h^2), \quad (24)$$

where $p(\sigma_b^2(h^2)|.)$ is the posterior conditional distribution given in (22) with $\sigma_b^2(h^2) = h^2/(1-h^2)$. We then use the Metropolis-Hastings algorithm to generate posterior samples for h^2. In particular, we use the independent random walk algorithm for h^2 with a Beta(2,8) distribution as the proposal distribution. With each sampled value of h^2, we can obtain a sampled value of $\sigma_b^2 = h^2/(1-h^2)$.

Sampling b

Finally, because of the relationship between **u** and **b** in equation (4), we can obtain the posterior conditional distribution for **b** as

$$p(\mathbf{b}|.) = \mathrm{MVN}_p\left(\frac{\sigma_b^2}{p}\mathbf{X}^T\mathbf{H}^{-1}(\mathbf{y} - \mathbf{W}\boldsymbol{\alpha} - \mathbf{X}\boldsymbol{\beta}), \sigma_b^2\sigma_e^2(p^{-1}\mathbf{I}_p - p^{-2}\sigma_b^2\mathbf{X}^T\mathbf{H}^{-1}\mathbf{X})\right), \quad (25)$$

where $\mathrm{MVN}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a p-dimensional multivariate normal distribution with mean $\boldsymbol{\mu}$ and variance-covariance $\boldsymbol{\Sigma}$. To reduce variance, we use the Rao-Blackwellised approximation to compute the mean of $\mathbf{b}$ at the end of the MCMC sampling, with

$$\hat{\mathbf{b}} = \frac{1}{p}\mathbf{X}^T \frac{1}{L}\sum_{\ell=1}^{L} (\sigma_b^2)^{(\ell)} (\mathbf{H}^{(\ell)})^{-1} (\mathbf{y} - \mathbf{W}\boldsymbol{\alpha}^{(\ell)} - \mathbf{X}\boldsymbol{\beta}^{(\ell)}). \quad (26)$$

where L is the total iterations of MCMC after burn in, ℓ denotes the posterior samples. These $\hat{\mathbf{b}}$ are added back to the posterior mean of $\boldsymbol{\beta}$ to yield the posterior mean of $\tilde{\boldsymbol{\beta}}$.

Efficient computation

We apply the algebra innovations recently developed for linear mixed models[1,8,9] to improve computational efficiency. Specifically, at the beginning of MCMC, we perform an eigen decomposition of $\mathbf{K} = \mathbf{U}\mathbf{D}\mathbf{U}^T$, where $\mathbf{U}$ is the matrix of eigenvectors and $\mathbf{D}$ is a diagonal matrix of eigenvalues[1,8,9]. Then we transform phenotype, genotypes and covariates as $\mathbf{U}^T\mathbf{y}$, $\mathbf{U}^T\mathbf{X}$, and $\mathbf{U}^T\mathbf{W}$. Afterwards, the likelihood conditional on the transformed variables become independent, thus alleviating much of the computational burden associated with the complex covariance structure resulted from the random effects $\mathbf{u}$.

The per-iteration computational cost of the above naive MCMC algorithm, after applying the linear mixed model algebra innovations, scales linearly both with the number of individuals and with the number of SNPs. Such computational cost can still be burdensome when we have millions of SNPs. To improve computation efficiency further, we develop a new, prioritized sampling strategy based on the recently developed random scan Gibbs sampler[10,11]. Specifically, we take advantage of the fact that for any complex traits, most SNPs have small effects (or are non-causal) while only a small proportion of SNPs have large effects (or are causal). The likely causal SNPs are important for phenotype prediction and their effect sizes need to be estimated accurately. In contrast, the likely non-causal SNPs often do not influence prediction performance much and their effect sizes individually do not require accurate estimation. Therefore, it is desirable to spend a large amount of computational resource on sampling likely causal SNPs to obtain accurate effect size estimates, while assigning a small amount of resource on sampling likely non-causal SNPs. Certainly, the above arguments are all conditional on a

fixed number of SNPs (i.e. spend extra computational resource on updating a fixed number of likely causal SNPs vs updating a fixed number of likely non-causal SNPs). To perform such prioritized sampling, we first obtain the top M marginally significant SNPs using LMM with the GEMMA algorithm. We treat these M selected SNPs as likely causal SNPs and update their effect sizes in each MCMC iteration. We then treat the unselected SNPs as likely non-causal SNPs and update their effect sizes once every T iterations. We set $M = 500$ and $T = 10$ (both are set to allow fast computation since the association signals are relatively strong in these two data) for the cattle and maize data, $M = 10^5$ and $T = 2$ (the two are set differently as the signals are relatively weak in this data) for the FHS data in the present study; for the GEUVADIS data we performed a full MCMC sampling as the small sample size there allows for efficient computation. Note that the choice M and T theoretically does not affect the stationary distribution, and we recommend exploring a few values of M and T in practice to achieve a balance between speed and accuracy. By prioritizing the computation resource on sampling the likely causal SNPs, our computational algorithm results in a dramatic reduction in computational cost, while yielding the same stationary distribution and maintaining the predictive performance of DPR. As an example, for the three traits MFP, MY and SCS in the cattle data, our naive MCMC takes approximately 25 hours to run 50,000 MCMC iterations. In contrast, our prioritized sampling algorithm reduces the computational cost down to approximately 5 hours, resulting in a five-fold speed improvement. The prediction performance of the prioritized sampling algorithm remains comparable with that of the naive MCMC: the resulting R^2 and MSE from the two algorithms were almost identical, with a correlation above 0.995 across 20 data splitting replicates. Note that the prioritizing sampling strategy we employ in DPR differs from the sample strategy used in BayesR[3], where a different set of M SNPs are used every T iteration. Indeed, our sampling strategy is still guaranteed to reach the same stationary distribution given a large number of iterations, regardless which set of M SNPs or which set of M and T values we choose to perform prioritized sampling.

Finally, we follow the truncated stick-breaking approximation approach of Blei and Jordan[12,13] and approximate the infinite normal mixture by a truncated normal mixture with K normal components. To ensure that π_k is well defined under the truncated

approximation (i.e. $\sum_{k=1}^{K}\pi_k = 1$), we set $v_k = 1$, $1 - v_k = 0$ for $k>K$[12,14,15]. With the truncated Dirichlet process approximation, we can draw posteriors via a simple Gibbs sampler, thus alleviating much of the computational burden associated with sampling the full Dirichlet process conditionally through the Chinese restaurant process. Because different truncated normal mixture approximations may result in different accuracy, we treat K as a parameter and use the deviance information criterion (DIC)[15-17] to select the optimal K automatically. To do so, we first perform MCMC sampling on a grid of K values from 2 to 10. For each K, we compute DIC using a small number of MCMC iterations (5,000). We select the optimal DPR model with the smallest DIC. We then run a large number of MCMC iterations (50,000) with the optimal DPR model. This strategy makes the selection of K in our DPR adaptive, while keeping computational cost in check. Note that this selection strategy may lead to local optimal and consequently hinders the performance of our method. Alternative and better strategies may improve DPR's prediction performance further. For the final 50,000 MCMC iterations, we discarded the first 10,000 as burn in and kept the remaining 40,000 for parameter estimation. We did not thin the MCMC chain[18], which may help improve prediction performance further. Finally, we also provided trace-plots for the log posterior likelihood of our model in all real data analyses following the recommendation in[15,19]. These trace-plots serve as a summary assessment of parameter convergence.

Mean Field Variational Inference for DPR

Despite the many algorithm innovations we use, the resulting MCMC algorithm is still computationally heavy. Therefore, we develop an alternative, much faster, algorithm based on variational Bayesian approximation[12,20-23]. Variational Bayesian approximation attempts to approximate the joint posterior distribution by a variational distribution, $q(\boldsymbol{\theta}) = \prod_j q(\theta_j)$, that assumes posterior independence among parameters θ_j. To do so, we minimize the Kullback-Leibler (KL) divergence between $p(\boldsymbol{\theta}\,|\,\mathbf{y})$ and $q(\boldsymbol{\theta})$

$$\begin{aligned} \mathrm{KL}(q(\boldsymbol{\theta})\,|\,p(\boldsymbol{\theta}\,|\,\mathbf{y})) &= E_{q(\boldsymbol{\theta})}(\log\frac{q(\boldsymbol{\theta})}{p(\boldsymbol{\theta}\,|\,\mathbf{y})}), \\ &= E_{q(\boldsymbol{\theta})}(\log q(\boldsymbol{\theta})) - E_{q(\boldsymbol{\theta})}(\log p(\boldsymbol{\theta},\mathbf{y})) + \log p(\mathbf{y}). \end{aligned} \tag{27}$$

Because the marginal probability $\log p(\mathbf{y})$ does not depend on the variational distribution, minimizing the KL divergence is equivalent to maximizing the evidence lower bound (ELBO)

$$E_{q(\boldsymbol{\theta})}(\log p(\boldsymbol{\theta},\mathbf{y})) - E_{q(\boldsymbol{\theta})}(\log q(\boldsymbol{\theta})). \tag{28}$$

To obtain the variational approximation, we can use the gradient ascent algorithm to maximize the above quantity with respect to each θ_j in turn. For each θ_j, we set the following derivative

$$\begin{aligned} &\frac{\partial E_{q(\boldsymbol{\theta})}(\log p(\boldsymbol{\theta},\mathbf{y})) - E_{q(\boldsymbol{\theta})}(\log q(\boldsymbol{\theta}))}{\partial q(\theta_j)} \\ &= \frac{\partial(\int q(\theta_j) E_{q(-\theta_j)}(\log p(\boldsymbol{\theta},\mathbf{y}))d\theta_j - \int q(\theta_j)\log q(\theta_j)d\theta_j)}{\partial q(\theta_j)} \\ &= E_{q(-\theta_j)}(\log p(\boldsymbol{\theta},\mathbf{y})) - \log q(\theta_j) - 1 \end{aligned} \tag{29}$$

to zero. Because $p(\boldsymbol{\theta},\mathbf{y})$ does not contain any parameter in $q(\theta_j)$, this leads to an update for each θ_j in the following form

$$q(\theta_j) \propto e^{E_{q(-\theta_j)}(\log p(\boldsymbol{\theta},\mathbf{y}))} \propto e^{E_{q(-\theta_j)}(\log p(\theta_j|\boldsymbol{\theta}_{-j},\mathbf{y}))}. \tag{30}$$

Inference based on the above factorized form of the variational distribution is commonly known as the mean field variational Bayesian approximation inference[20,21,23-25].

We apply the mean field variational Bayesian approximation to DPR. Because computing ELBO is difficult for non-analytic variational distributions[26,27], we cannot integrate out $\mathbf{u}$ from model (5) as we do for MCMC. Instead, we keep $\mathbf{u}$. We also denote $\mathbf{g} = \mathbf{U}^T\mathbf{u}$. Our joint log posterior is

$$
\begin{aligned}
\log p(\boldsymbol{\theta},\mathbf{y}) &= \log p(\mathbf{y}\mid\boldsymbol{\alpha},\boldsymbol{\beta},\mathbf{u}) + \log p(\boldsymbol{\beta}\mid\boldsymbol{\gamma},\sigma_k^2,\sigma_e^2) + \log p(\mathbf{u}\mid\sigma_b^2,\sigma_e^2) \\
&\quad + \log p(\boldsymbol{\gamma}\mid\nu_k) + \log p(\nu_k\mid\lambda) + \log p(\sigma_k^2\mid a_{0k},b_{0k}) + \log p(\sigma_e^2\mid a_{0e},b_{0e}) \\
&\quad + \log p(\sigma_b^2\mid a_{0b},b_{0b}) + \log p(\lambda\mid a_{0\lambda},b_{0\lambda}) \\
&= C - \frac{n}{2}\log(\sigma_e^2) - \frac{1}{2\sigma_e^2}(\mathbf{y}-\mathbf{W}\boldsymbol{\alpha}-\mathbf{X}\boldsymbol{\beta}-\mathbf{u})^T(\mathbf{y}-\mathbf{W}\boldsymbol{\alpha}-\mathbf{X}\boldsymbol{\beta}-\mathbf{u}) \\
&\quad + \sum_i\sum_{k=2}^{\infty}\gamma_{ik}\left(-\frac{1}{2}\log(\sigma_e^2) - \frac{1}{2}\log(\sigma_k^2) - \frac{\beta_{ik}^2}{2\sigma_k^2\sigma_e^2}\right) \\
&\quad - \frac{n}{2}\log(\sigma_e^2) - \frac{n}{2}\log(\sigma_b^2) - \frac{1}{2}\log|\mathbf{K}| - \frac{1}{2}\mathbf{u}^T(\sigma_e^2\sigma_b^2\mathbf{K})^{-1}\mathbf{u} \\
&\quad + \sum_i\sum_{k=1}^{\infty}\gamma_{ik}\left(\log(\nu_k) + \sum_{l=1}^{k-1}\log(1-\nu_l)\right) + \sum_k^{\infty}((\lambda-1)\log(1-\nu_k)+\log(\lambda)) \\
&\quad - \sum_k^{\infty}(a_{0k}+1)\log(\sigma_k^2) - \sum_k^{\infty}b_{0k}\sigma_k^{-2} - (a_{0e}+1)\log(\sigma_e^2) - b_{0e}\sigma_e^{-2} \\
&\quad - (a_{0b}+1)\log(\sigma_b^2) - b_{0b}\sigma_b^{-2} + (a_{0\lambda}-1)\log(\lambda) - b_{0\lambda}\lambda, \qquad (31)
\end{aligned}
$$

where again C is a normalizing constant. We will ignore the constant terms in the following updates.

We follow the truncated stick-breaking approximation approach of Blei and Jordan[12] and use a finite mixture with a fixed number of normal components, K, as an approximation to the posterior distribution. The parameter K here is considered as a variational parameter and we choose K by optimizing ELBO. Note again that although we use a finite mixture as an approximation to the posterior distribution, our likelihood still consists of a mixture of infinitely many normal distributions[12]. To choose K, we perform variational inference with DPR on different K values ranging from 2 to 10. Following[12], we then choose the optimal DPR model with the largest ELBO and we present results based on the optimal DPR.

Variational distribution for α_j

First, for α_j, we have

$$
\begin{aligned}
\log q(\alpha_j) &= -\frac{E(\sigma_e^{-2})\mathbf{w}_j{}^T\mathbf{w}_j}{2}\alpha_j^2 \\
&\quad + E(\sigma_e^{-2})\mathbf{w}_j{}^T\Big(\mathbf{y} - \sum_{m\neq j}\mathbf{w}_m E(\alpha_m) - \mathbf{X}E(\boldsymbol{\beta}) - E(\mathbf{u})\Big)\alpha_j. \qquad (32)
\end{aligned}
$$

Therefore, the variation distribution for α_j is $q(\alpha_j) = N(m_j, s_j^2)$, where

$$m_j = \frac{\mathbf{w}_j{}^T(\mathbf{y} - \sum_{m \neq j}\mathbf{w}_m E(\alpha_m) - \mathbf{X}E(\boldsymbol{\beta}) - E(\mathbf{u}))}{\mathbf{w}_j{}^T\mathbf{w}_j},$$
$$s_j^2 = \frac{E(\sigma_e^{-2})^{-1}}{\mathbf{w}_j{}^T\mathbf{w}_j}. \tag{33}$$

Variational distributions for β_{ik} and γ_{ik}

For β_{ik} and γ_{ik}, we have

$$\begin{aligned}\log q(\beta_{ik}, \gamma_{ik}) = &-\frac{E(\sigma_e^{-2})\mathbf{x}_i^T\mathbf{x}_i}{2}E(\beta_i^2) \\ &+E(\sigma_e^{-2})\mathbf{x}_i^T(\mathbf{y} - \mathbf{W}E(\boldsymbol{\alpha}) - \sum_{m \neq i}\mathbf{x}_m E(\beta_m) - E(\mathbf{u}))\beta_i \\ &+\gamma_{ik}(-\frac{1}{2}\log E(\sigma_e^2) - \frac{1}{2}\log E(\sigma_k^2) - \frac{1}{2}E(\sigma_k^{-2})E(\sigma_e^{-2})\beta_{ik}^2) \\ &+\gamma_{ik}(\log E(\nu_k) + \sum_{l=1}^{k-1}\log E(1-\nu_l)).\end{aligned} \tag{34}$$

A natural update form for $q(\beta_{ik}, \gamma_{ik})$ is thus

$$\begin{aligned}q(\beta_{ik} \mid \gamma_{ik} = 1) &= N(m_{ik}, s_{ik}^2), \\ q(\gamma_{ik} = 1) &= \varphi_{ik} \propto e^{m_{ik}^2/2s_{ik}^2 + \log(s_{ik}) - E(\log(\sigma_e)) - E(\log(\sigma_k)) + E(\log(\nu_k)) + \sum_{l=1}^{k-1}E(\log(1-\nu_l))},\end{aligned} \tag{35}$$

where

$$m_{ik} = \frac{\mathbf{x}_i{}^T(\mathbf{y} - \mathbf{W}E(\boldsymbol{\alpha}) - \sum_{m \neq i}\mathbf{x}_m E(\beta_m) - E(\mathbf{u}))}{\mathbf{x}_i{}^T\mathbf{x}_i + E(\sigma_k^{-2})},$$
$$s_{ik}^2 = \frac{E(\sigma_e^{-2})^{-1}}{\mathbf{x}_i{}^T\mathbf{x}_i + E(\sigma_k^{-2})}. \tag{36}$$

Variational distribution for v

For *v,* we have

$$\log q(\nu_k) = \sum_i E(\gamma_{ik})\log(\nu_k) + \sum_i\sum_{l=k+1}^{\infty}E(\gamma_{il})\log(1-\nu_k) + (E(\lambda) - 1)\log(1-\nu_k). \tag{37}$$

Thus $q(\nu_k) = \text{Beta}(\kappa_k, \lambda_k)$, where

$$\kappa_k = \sum_i E(\gamma_{ik}) + 1,$$
$$\lambda_k = \sum_i \sum_{l=k+1}^{\infty} E(\gamma_{il}) + E(\lambda). \tag{38}$$

Variational distribution for σ_k^2

For σ_k^2, we have

$$\log q(\sigma_k^2) = -(\frac{\sum_i E(\gamma_{ik})}{2} + a_{0k} + 1)\log(\sigma_k^2) - (\frac{\sum_i E(\gamma_{ik}\beta_{ik}^2)E(\sigma_e^{-2})}{2} + b_{0k})\sigma_k^{-2}. \tag{39}$$

Thus $q(\sigma_k^2) = \text{inverse-gamma}(a_k, b_k)$, where

$$a_k = \frac{1}{2}\sum_i E(\gamma_{ik}) + a_{0k},$$
$$b_k = \frac{1}{2}\sum_i E(\gamma_{ik}\beta_{ik}^2)E(\sigma_e^{-2}) + b_{0k}. \tag{40}$$

Variational distribution for λ

For λ, we have

$$\log q(\lambda) = \lambda(\sum_k^{\infty} \log E(1-\nu_k) - b_{0\lambda}) + \log(\lambda)(a_{0\lambda} + \sum_k^{\infty} 1_k). \tag{41}$$

Thus $q(\lambda) = \text{gamma}(a_\lambda, b_\lambda)$, where

$$a_\lambda = a_{0\lambda} + \sum_k^{\infty} 1_k,$$
$$b_\lambda = b_{0\lambda} - \sum_k^{\infty} \log E(1-\nu_k). \tag{42}$$

Variational distribution for g

For $\mathbf{g}$, we have

$$\log q(\mathbf{g}) = -\frac{1}{2\sigma_e^2}(\mathbf{U}^T\mathbf{y} - \mathbf{U}^T\mathbf{W}E(\boldsymbol{\alpha}) - \mathbf{U}^T\mathbf{X}E(\boldsymbol{\beta}) - \mathbf{g})^T(\mathbf{U}^T\mathbf{y} - \mathbf{U}^T\mathbf{W}E(\boldsymbol{\alpha}) - \mathbf{U}^T\mathbf{X}E(\boldsymbol{\beta}) - \mathbf{g})$$
$$-\frac{1}{2}\mathbf{g}^T(\sigma_e^2\sigma_b^2\mathbf{D})^{-1}\mathbf{g}. \tag{43}$$

Thus $q(\mathbf{g}) = \text{MVN}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where

$$\boldsymbol{\mu} = (E(\sigma_b^{-2})\mathbf{D}^{-1}) + \mathbf{I}_n)^{-1}(\mathbf{U}^T\mathbf{y} - \mathbf{U}^T\mathbf{W}E(\boldsymbol{\alpha}) - \mathbf{U}^T\mathbf{X}E(\boldsymbol{\beta})),$$
$$\boldsymbol{\Sigma} = (E(\sigma_b^{-2})\mathbf{D}^{-1}) + \mathbf{I}_n)^{-1} E(\sigma_e^{-2})^{-1}. \tag{44}$$

Here, the covariance matrix is diagonal, which facilitates computation. As in MCMC, we use the relationship in equation (4) to obtain the mean of **b** at the end of the algorithm. The estimated mean of **b** is added back to the mean of $\boldsymbol{\beta}$ to obtain a mean estimate for $\tilde{\boldsymbol{\beta}}$.

Variational distribution for σ_b^2

For σ_b^2, we have

$$\log q(\sigma_b^2) = -\frac{n}{2}\log(\sigma_b^2) - \frac{1}{2}\sum_i \sigma_b^{-2} E(g_i)^2 / E(d_i\sigma_e^2) - (a_{0b}+1)\log(\sigma_b^2) - b_{0b}\sigma_b^{-2}, \tag{45}$$

where d_i is the ith diagonal element of **D**. Thus $q(\sigma_b^2) = \text{inverse-gamma}(a_b, b_b)$, where

$$a_b = \frac{n}{2} + a_{0b},$$
$$b_b = \frac{1}{2}\sum_i E(g_i)^2 E(d_i^{-1}\sigma_e^{-2}) + b_{0b}. \tag{46}$$

Variational distribution for σ_e^2

Finally, for σ_e^2, we have

$$\log q(\sigma_e^2) = -(n + \sum_i\sum_{k=2} E(\gamma_{ik})/2 + a_{0e} + 1)\log(\sigma_e^2) - \frac{1}{2}A\times\sigma_e^{-2}$$
$$-\frac{1}{2}(\sum_i\sum_k E(\gamma_{ik}\beta_{ik}^2)E(\sigma_k^{-2}) + \sum_i E(g_i)^2 / E(d_i\sigma_b^2) + 2b_{0e})\sigma_e^{-2}. \tag{47}$$

Thus $q(\sigma_e^2) = \text{inverse-gamma}(a_e, b_e)$, where

$$a_e = n + \sum_i\sum_{k=2} E(\gamma_{ik})/2 + a_{0e},$$
$$b_e = \frac{1}{2}(A + \sum_i\sum_{k=2} E(\gamma_{ik}\beta_{ik}^2)E(\sigma_k^{-2}) + \sum_i E(g_i)^2 E(\sigma_b^{-2}d_i^{-1}) + 2b_{0e}),$$
$$A = (\mathbf{U}^T\mathbf{y} - \mathbf{U}^T\mathbf{W}E(\boldsymbol{\alpha}) - \mathbf{U}^T\mathbf{X}E(\boldsymbol{\beta}) - E(\mathbf{g}))^T(\mathbf{U}^T\mathbf{y} - \mathbf{U}^T\mathbf{W}E(\boldsymbol{\alpha}) - \mathbf{U}^T\mathbf{X}E(\boldsymbol{\beta}) - E(\mathbf{g}))$$
$$+\sum_j \mathbf{w}_j^T\mathbf{w}_j s_j^2 + \sum_i \boldsymbol{\Sigma}_{ii} + \sum_i \mathbf{x}_i^T\mathbf{x}_i(\sum_k E(\gamma_{ik})(m_{ik}^2 + s_{ik}^2) - (\sum_k E(\gamma_{ik})m_{ik})^2), \tag{48}$$

where $\boldsymbol{\Sigma}_{ii}$ is the ith diagonal element of $\boldsymbol{\Sigma}$ given in (44).

To evaluate all the above expectations, we need

$$
\begin{aligned}
E_{q(\nu_k)}(\log(\nu_k)) &= \Psi(\kappa_k) - \Psi(\kappa_k + \lambda_k), \\
E_{q(\nu_k)}(\log(1-\nu_k)) &= \Psi(\lambda_k) - \Psi(\kappa_k + \lambda_k), \\
E_{q(\gamma_i,\tilde{\beta}_i)}(\gamma_i \beta_i^2) &= \sum_k \varphi_{ik}(m_{ik}^2 + s_{ik}^2), \\
E_{q(\gamma_i,\tilde{\beta}_i)}(\beta_i) &= \sum_k \varphi_{ik} m_{ik}, \\
E_{q(\alpha_j)}(\alpha_j^2) &= m_j^2 + s_j^2, \\
E_{q(\alpha_j)}(\alpha_j) &= m_j, \\
E(\mathbf{g}) &= \boldsymbol{\mu}, \\
E_{q(\sigma_k^2)}(\log \sigma_k) &= \frac{1}{2}(\log(b_k) - \Psi(a_k)), \\
E_{q(\sigma_k^2)}(\sigma_k^{-2}) &= \frac{a_k}{b_k}, \\
E_{q(\lambda)}(\log \lambda) &= \Psi(a_\lambda) - \log(b_\lambda), \\
E_{q(\lambda)}(\lambda) &= a_\lambda / b_\lambda,
\end{aligned}
\tag{49}
$$

where Ψ is the digamma function.

ELBO and convergence

We use ELBO to check convergence of the variational algorithm. For the explicit form of ELBO, first, we have

$$
\begin{aligned}
E_{q(\beta_i,\gamma_i)}(\log(q(\beta_i,\gamma_i))) &= \sum_{k=2} \varphi_{ik}(\log \varphi_{ik} - \frac{1}{2}\log(2\pi \times e \times s_{ik}^2) - \frac{1}{2}), \\
E_{q(\alpha_j)}(\log(q(\alpha_j))) &= -\frac{1}{2}\log(s_j^2), \\
E_{q(g_i)}(\log(q(g_i))) &= -\frac{1}{2}\log(\boldsymbol{\Sigma}_{ii}), \\
E_{q(\nu_k)}(\log(q(\nu_k))) &= \log\Gamma(\kappa_k + \lambda_k) - \log\Gamma(\kappa_k) - \log\Gamma(\lambda_k) \\
&\quad + (\kappa_k - 1)(\Psi(\kappa_k) - \Psi(\kappa_k + \lambda_k)) \\
&\quad + (\lambda_k - 1)(\Psi(\lambda_k) - \Psi(\kappa_k + \lambda_k)), \\
E_{q(\sigma_k^2)}(\log(q(\sigma_k^2))) &= a_k \log b_k - \log\Gamma(a_k) + (a_k + 1)(\Psi(a_k) - \log b_k) - a_k, \\
E_{q(\sigma_e^2)}(\log(q(\sigma_e^2))) &= a_e \log b_e - \log\Gamma(a_e) + (a_e + 1)(\Psi(a_e) - \log b_e) - a_e, \\
E_{q(\sigma_b^2)}(\log(q(\sigma_b^2))) &= a_b \log b_b - \log\Gamma(a_b) + (a_b + 1)(\Psi(a_b) - \log b_b) - a_b, \\
E_{q(\lambda)}(\log(q(\lambda))) &= \log b_\lambda - \log\Gamma(a_\lambda) - (1 - a_\lambda)\Psi(a_\lambda) - a_\lambda.
\end{aligned}
\tag{50}
$$

In addition, we have

$$
\begin{aligned}
E_{q(\boldsymbol{\theta})}(\log p(\boldsymbol{\theta},\mathbf{y})) = &-(a_e+1)(\log b_e - \Psi(a_e)) - \frac{1}{2}\sum_i\sum_{k=2}\varphi_{ik}(\log b_k - \Psi(a_k)) \\
&-(a_{0k}+1)\sum_{k=2}(\log b_k - \Psi(a_k)) - (a_b+1)(\log b_b - \Psi(a_b)) \\
&-\frac{1}{2}\frac{a_e}{b_e}(A + \sum_i\sum_{k=2}\varphi_{ik}\frac{a_k}{b_k}(m_{ik}^2 + s_{ik}^2) + \frac{a_b}{b_b}\sum_i(\mu_i^2 + \boldsymbol{\Sigma}_{ii})/d_i + 2b_{0e}) \\
&+\sum_i\sum_{k=1}\varphi_{ik}(\Psi(\kappa_k) - \Psi(\kappa_k+\lambda_k) + \sum_{l=1}^{k-1}(\Psi(\lambda_l) - \Psi(\kappa_l+\lambda_l))) \\
&+(\frac{a_\lambda}{b_\lambda} - 1)(\sum_k(\Psi(\lambda_k) - \Psi(\kappa_k+\lambda_k))) + (a_\lambda - 1)(\Psi(a_\lambda) - \log b_\lambda) \\
&-b_{0k}\sum_{k=2}\frac{a_k}{b_k} - b_{0b}\frac{a_b}{b_b} - b_{0\lambda}\frac{a_\lambda}{b_\lambda}.
\end{aligned} \tag{51}
$$

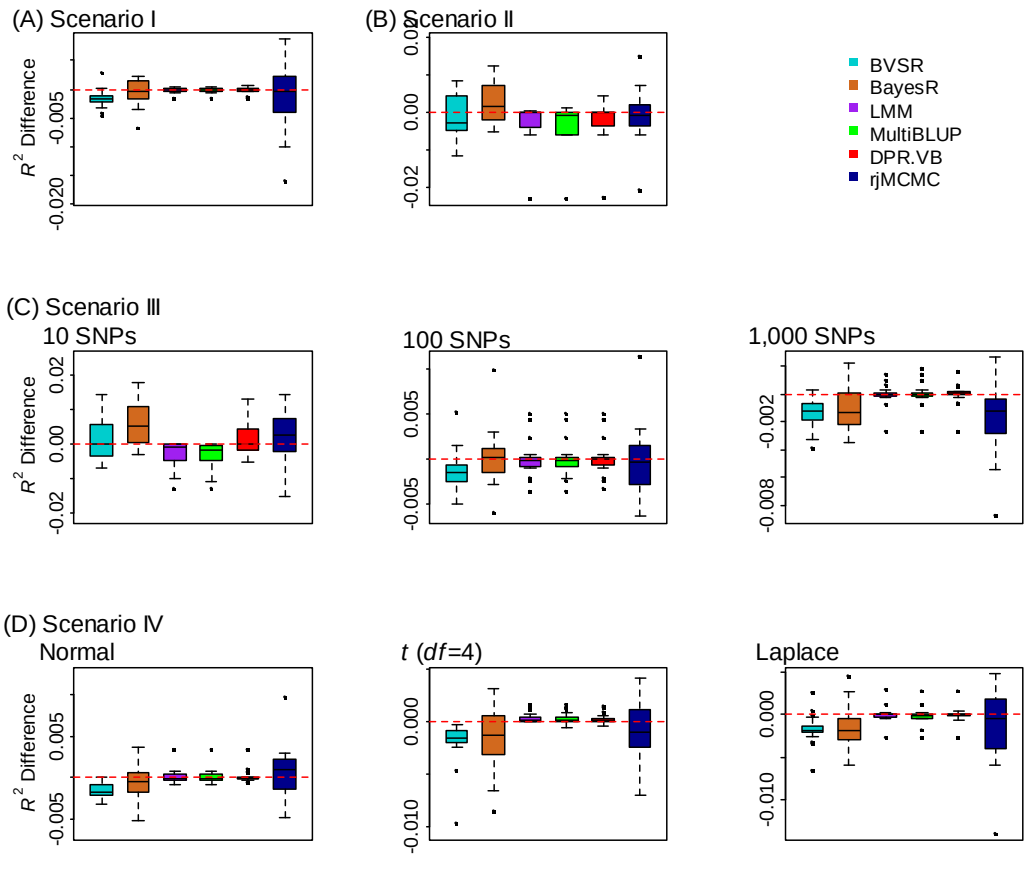
Finally,

$$
\begin{aligned}
E_{q(\boldsymbol{\theta})}(\log(q(\boldsymbol{\theta})) = &-(a_e+1)(\log b_e - \Psi(a_e)) - a_e \\
&-\sum_k(a_k+1)(\log b_k - \Psi(a_k)) \\
&-(a_b+1)(\log b_b - \Psi(a_b)) \\
&+\sum_i\sum_{k=1}\varphi_{ik}(\Psi(\kappa_k) - \Psi(\kappa_k+\lambda_k) + \sum_{l=1}^{k-1}(\Psi(\lambda_l) - \Psi(\kappa_l+\lambda_l))) \\
&+(\frac{a_\lambda}{b_\lambda} - 1)(\sum_k(\Psi(\lambda_k) - \Psi(\kappa_k+\lambda_k))) + (a_\lambda - 1)(\Psi(a_\lambda) - \log b_\lambda).
\end{aligned} \tag{52}
$$

Therefore, the ELBO is

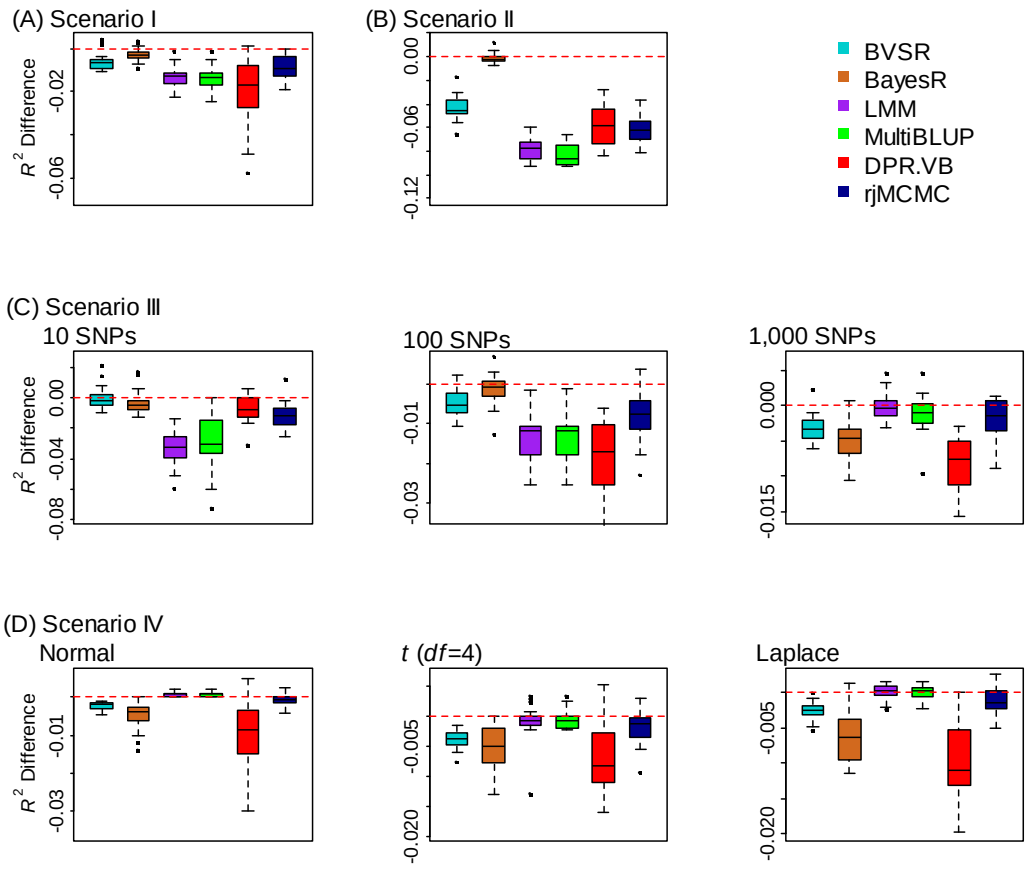
$$
\begin{aligned}
\text{ELBO} &= E_{q(\boldsymbol{\theta})}(\log p(\boldsymbol{\theta},\mathbf{y})) - E_{q(\boldsymbol{\theta})}(\log(q(\boldsymbol{\theta})) \\
&= \log\Gamma(a_e) - a_e\log b_e \\
&\quad + \log\Gamma(a_b) - a_b\log b_b + a_b \\
&\quad + \sum_{k=2}(\log\Gamma(a_k) - a_k\log b_k + a_k) \\
&\quad + \sum_k(\log\Gamma(k_k) + \log\Gamma(\lambda_k) - \log\Gamma(k_k+\lambda_k)) \\
&\quad - \sum_i\sum_{k=2}\varphi_{ik}(\log\varphi_{ik} - \frac{1}{2}\log(2\pi\times e\times s_{ik}^2) - \frac{1}{2}) + \frac{1}{2}\sum_j\log(s_j^2) + \frac{1}{2}\sum_i\log(\boldsymbol{\Sigma}_{ii}) \\
&\quad + \log\Gamma(a_\lambda) - a_\lambda\log b_\lambda + a_\lambda - b_{0k}\sum_{k=2}\frac{a_k}{b_k} - b_{0b}\frac{a_b}{b_b} - b_{0\lambda}\frac{a_\lambda}{b_\lambda}.
\end{aligned} \tag{53}
$$

Supplementary Figures and Tables



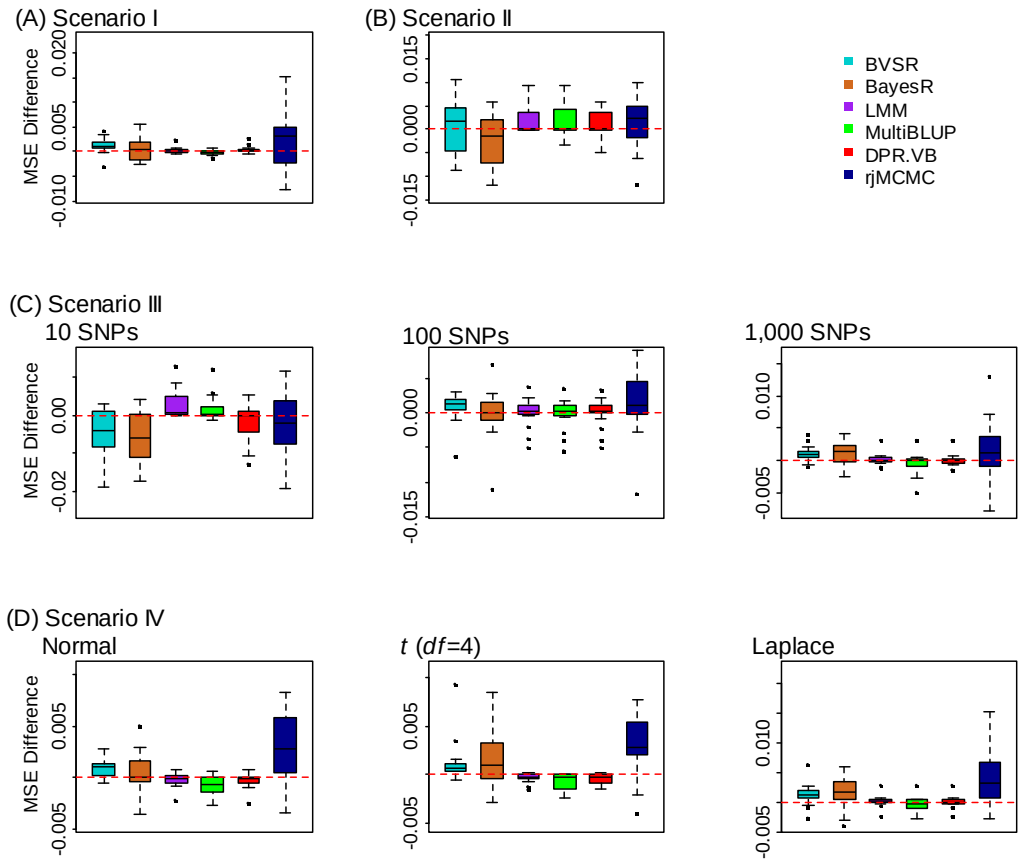
Supplementary Figure 1. Comparison of prediction performance of several methods with DPR.MCMC in simulations when PVE=0.2. Performance is measured by R^2 difference with respect to DPR.MCMC, where a negative value (i.e. values below the red horizontal line) indicates worse performance than DPR.MCMC. The sample R^2 differences are obtained from 20 replicates in each scenario. Methods for comparison include BVSR (cyan), BayesR (chocolate), LMM (purple), MultiBLUP (green), DPR.VB (red), rjMCMC (black blue) and DPR.MCMC. Simulation scenarios include: (A) Scenario I, which satisfies the DPR modeling assumption; (B) Scenario II, which satisfies the BayesR modeling assumption; (C) Scenario III, where the number of SNPs in the large effect group is 10, 100, or 1,000; and (D) Scenario IV, where the effect sizes are generated from either a normal distribution, a t-distribution or a Laplace distribution. For each box plot, the bottom and top of the box are the first and third quartiles, while the

ends of whiskers represent either the lowest datum within 1.5 interquartile range of the lower quartile or the highest datum within 1.5 interquartile range of the upper quartile. For DPR.MCMC, the mean predictive R^2 in the test set and the standard deviation for the eight settings are respectively 0.074 (0.020), 0.081 (0.016), 0.076 (0.018), 0.072 (0.019), 0.064 (0.016), 0.083 (0.023), 0.077 (0.016) and 0.077 (0.017).



Supplementary Figure 2. Comparison of prediction performance of several methods with DPR.MCMC in simulations when PVE=0.8. Performance is measured by R^2 difference with respect to DPR.MCMC, where a negative value (i.e. values below the red horizontal line) indicates worse performance than DPR.MCMC. The sample R^2 differences are obtained from 20 replicates in each scenario. Methods for comparison include BVSR (cyan), BayesR (chocolate), LMM (purple), MultiBLUP (green), DPR.VB (red), rjMCMC (black blue) and DPR.MCMC. Simulation scenarios include: (A) Scenario I, which satisfies the DPR modeling assumption; (B) Scenario II, which satisfies the BayesR modeling assumption; (C) Scenario III, where the number of SNPs in the large effect group is 10, 100, or 1,000; and (D) Scenario IV, where the effect sizes are generated from either a normal distribution, a t-distribution or a Laplace distribution. For each box plot, the bottom and top of the box are the first and third quartiles, while the ends of whiskers represent either the lowest datum within 1.5 interquartile range of the lower quartile or the highest datum within 1.5 interquartile range of the upper quartile.

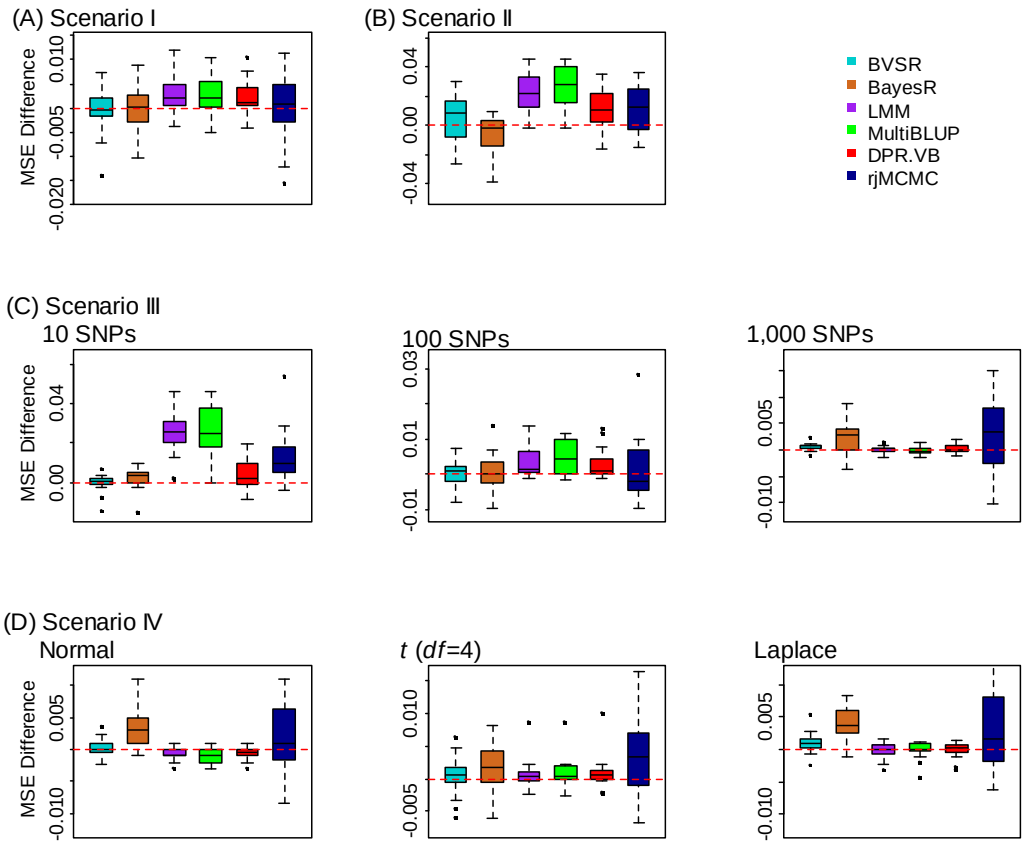
For DPR.MCMC, the mean predictive R^2 in the test set and the standard deviation for the eight settings are respectively 0.554 (0.028), 0.622 (0.022), 0.569 (0.023), 0.548 (0.027), 0.537 (0.030), 0.543 (0.028), 0.546 (0.027) and 0.539 (0.022).



383

384 **Supplementary Figure 3. Comparison of prediction performance of several methods**
385 **with DPR.MCMC in simulations when PVE=0.2.** Performance is measured by MSE
386 difference with respect to DPR.MCMC, where a positive value (i.e. values above the red
387 horizontal line) indicates worse performance than DPR.MCMC. The sample MSE
388 differences are obtained from 20 replicates in each scenario. Methods for comparison
389 include BVSR (cyan), BayesR (chocolate), LMM (purple), MultiBLUP (green), DPR.VB
390 (red), rjMCMC (black blue) and DPR.MCMC. Simulation scenarios include: (A)
391 Scenario I, which satisfies the DPR modeling assumption; (B) Scenario II, which
392 satisfies the BayesR modeling assumption; (C) Scenario III, where the number of SNPs
393 in the large effect group is 10, 100, or 1,000; and (D) Scenario IV, where the effect sizes
394 are generated from either a normal distribution, a t-distribution or a Laplace distribution.
395 For each box plot, the bottom and top of the box are the first and third quartiles, while the
396 ends of whiskers represent either the lowest datum within 1.5 interquartile range of the
397 lower quartile or the highest datum within 1.5 interquartile range of the upper quartile.

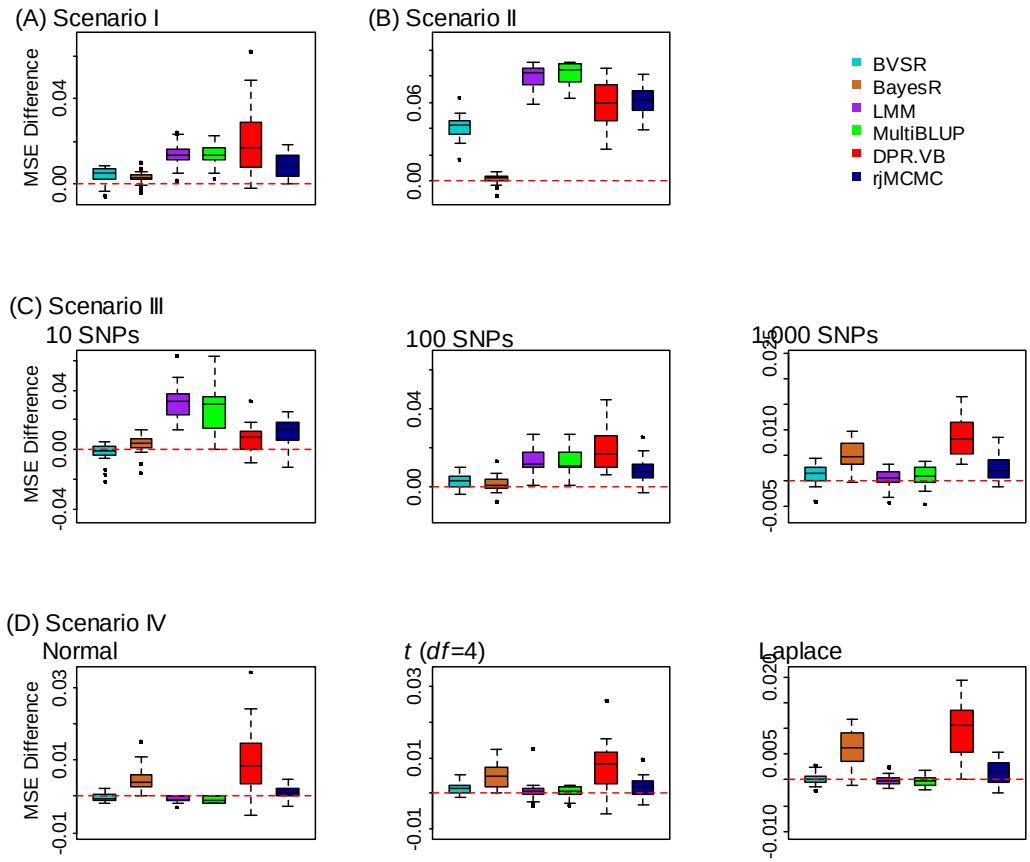
For DPR.MCMC, the mean predictive MSE in the test set and the standard deviation for the eight settings are respectively 0.919 (0.044), 0.910 (0.038), 0.929 (0.036), 0.944 (0.053), 0.923 (0.038), 0.925 (0.033), 0.924 (0.037) and 0.918 (0.037).



410

411 **Supplementary Figure 4. Comparison of prediction performance of several methods**
412 **with DPR.MCMC in simulations when PVE=0.5.** Performance is measured by MSE
413 difference with respect to DPR.MCMC, where a positive value (i.e. values above the red
414 horizontal line) indicates worse performance than DPR.MCMC. The sample MSE
415 differences are obtained from 20 replicates in each scenario. Methods for comparison
416 include BVSR (cyan), BayesR (chocolate), LMM (purple), MultiBLUP (green), DPR.VB
417 (red), rjMCMC (black blue) and DPR.MCMC. Simulation scenarios include: (A)
418 Scenario I, which satisfies the DPR modeling assumption; (B) Scenario II, which
419 satisfies the BayesR modeling assumption; (C) Scenario III, where the number of SNPs
420 in the large effect group is 10, 100, or 1,000; and (D) Scenario IV, where the effect sizes
421 are generated from either a normal distribution, a t-distribution or a Laplace distribution.
422 For each box plot, the bottom and top of the box are the first and third quartiles, while the
423 ends of whiskers represent either the lowest datum within 1.5 interquartile range of the
424 lower quartile or the highest datum within 1.5 interquartile range of the upper quartile.

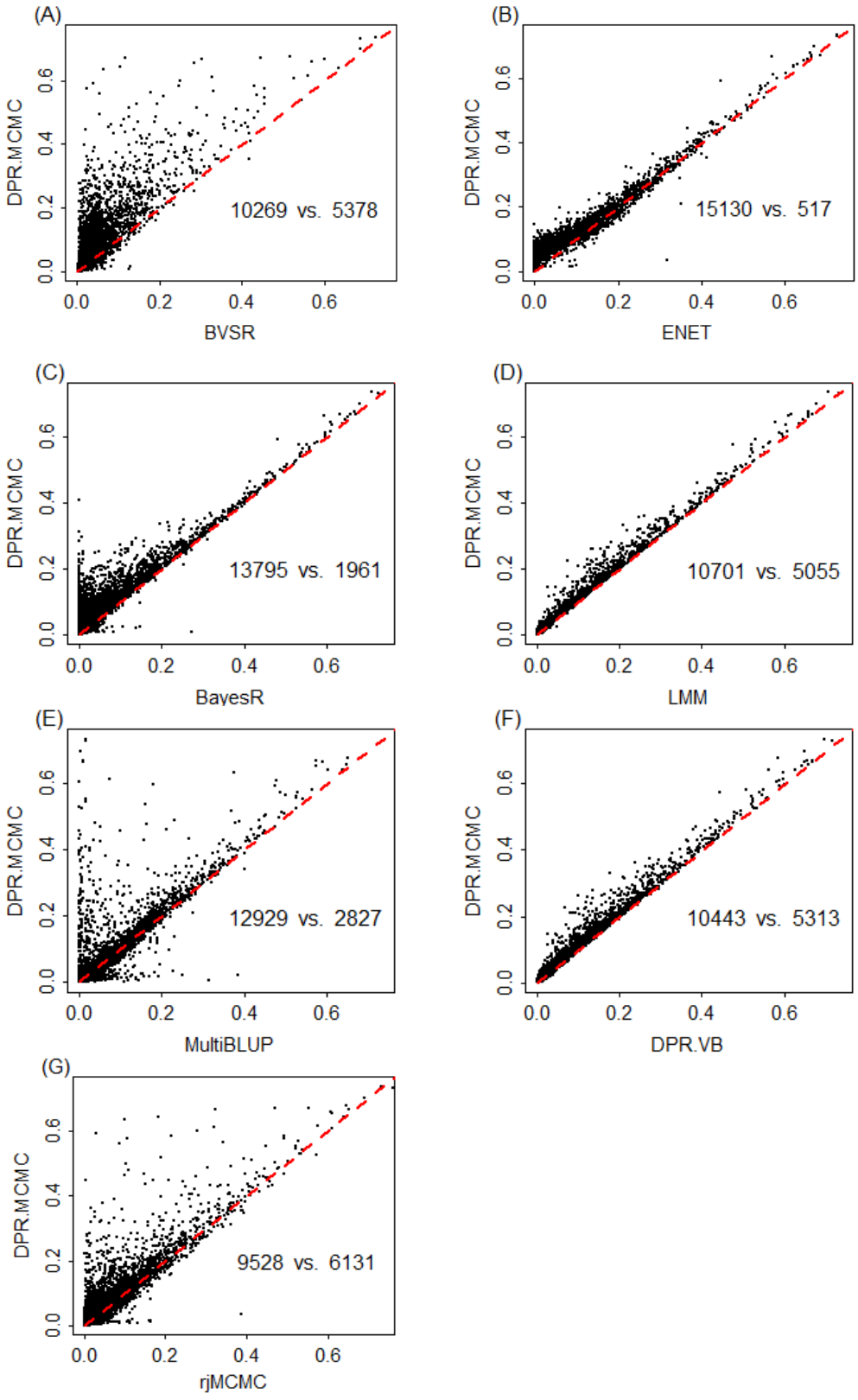
For DPR.MCMC, the mean predictive MSE in the test set and the standard deviation for the eight settings are respectively 0.722 (0.043), 0.701 (0.028), 0.707 (0.034), 0.717 (0.037), 0.727 (0.034), 0.734 (0.040), 0.721 (0.032) and 0.720 (0.028).



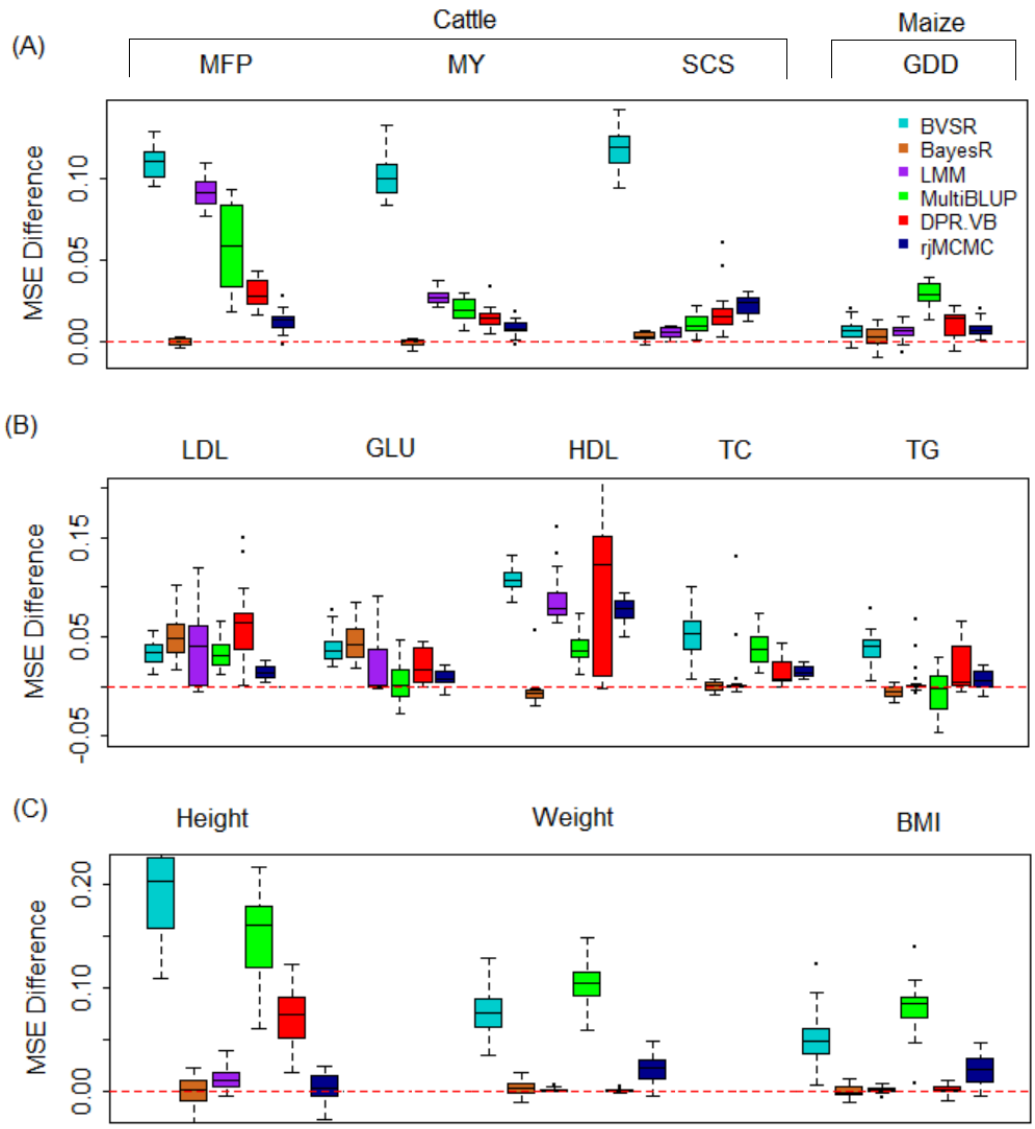
438

439 **Supplementary Figure 5. Comparison of prediction performance of several methods**
440 **with DPR.MCMC in simulations when PVE=0.8.** Performance is measured by MSE
441 difference with respect to DPR.MCMC, where a positive value (i.e. values above the red
442 horizontal line) indicates worse performance than DPR.MCMC. The sample MSE
443 differences are obtained from 20 replicates in each scenario. Methods for comparison
444 include BVSR (cyan), BayesR (chocolate), LMM (purple), MultiBLUP (green), DPR.VB
445 (red), rjMCMC (black blue) and DPR.MCMC. Simulation scenarios include: (A)
446 Scenario I, which satisfies the DPR modeling assumption; (B) Scenario II, which
447 satisfies the BayesR modeling assumption; (C) Scenario III, where the number of SNPs
448 in the large effect group is 10, 100, or 1,000; and (D) Scenario IV, where the effect sizes
449 are generated from either a normal distribution, a t-distribution or a Laplace distribution.
450 For each box plot, the bottom and top of the box are the first and third quartiles, while the
451 ends of whiskers represent either the lowest datum within 1.5 interquartile range of the
452 lower quartile or the highest datum within 1.5 interquartile range of the upper quartile.

For DPR.MCMC, the mean predictive MSE in the test set and the standard deviation for the eight settings are respectively 0.443 (0.032), 0.379 (0.016), 0.429 (0.024), 0.454 (0.023), 0.464 (0.030), 0.465 (0.027), 0.454 (0.032) and 0.457 (0.022).


(A)
DPR.MCMC
BVSR
10269 vs. 5378
(B)
ENET
15130 vs. 517
(C)
BayesR
13795 vs. 1961
(D)
LMM
10701 vs. 5055
(E)
MultiBLUP
12929 vs. 2827
(F)
DPR.VB
10443 vs. 5313
(G)
rjMCMC
9528 vs. 6131
0.0
0.2
0.4
0.6

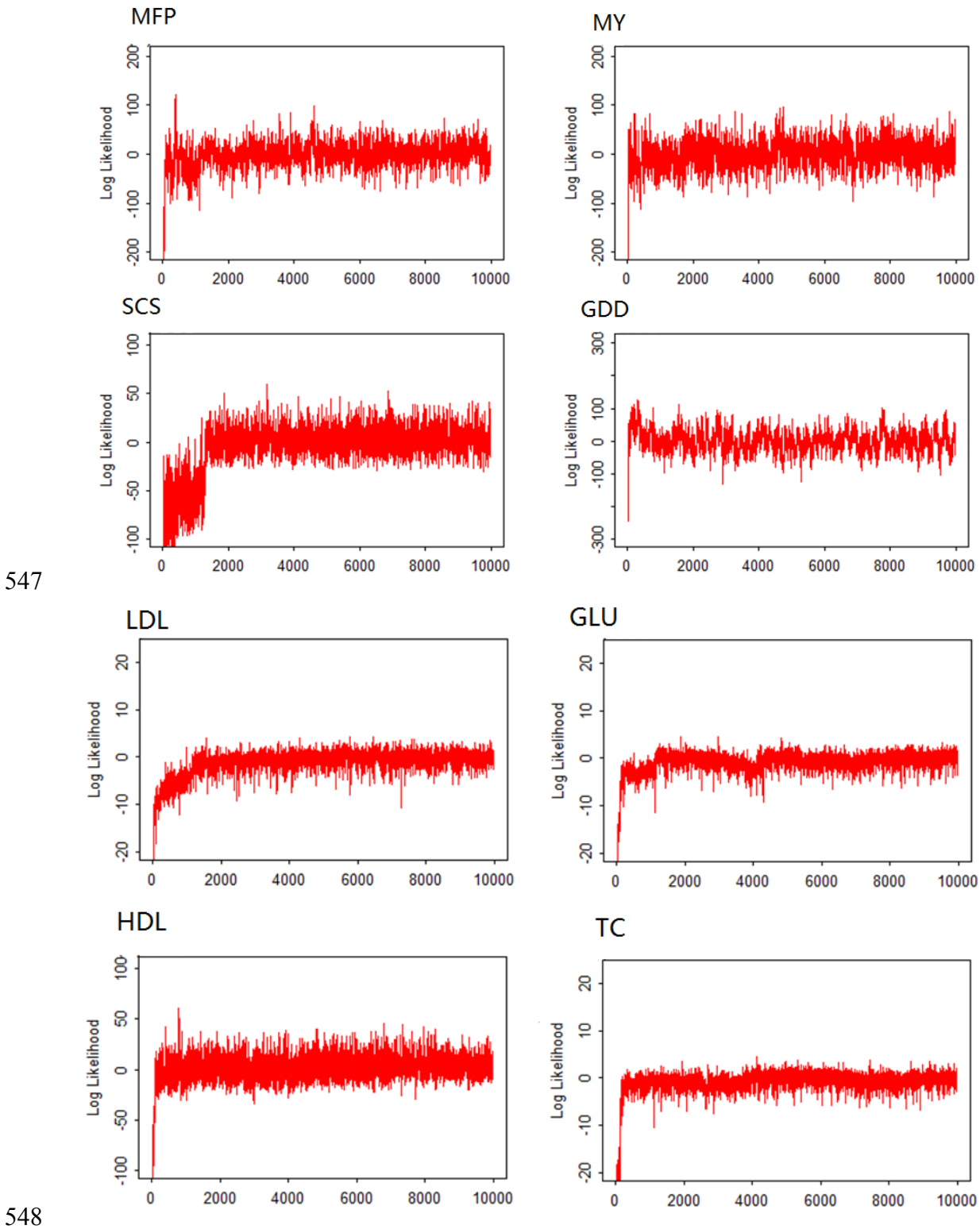
Supplementary Figure 6. Comparison of predictive R^2 from DPR.MCMC with the other six methods for predicting gene expression levels in the GEUVADIS data. Scatter plots show (A) predictive R^2 in the test data obtained by DPR.MCMC vs that obtained by BVSR for all genes; (B) DPR.MCMC vs ENET; (C) DPR.MCMC vs BayesR; (D) DPR.MCMC vs LMM; (E) DPR.MCMC vs MultiBLUP; (F) DPR.MCMC vs DPR.VB; (G) DPR.MCMC vs rjMCMC. Each panel also lists the number of genes where DPR.MCMC performs better (first number) and the number of genes where DPR.MCMC performs worse (second number).

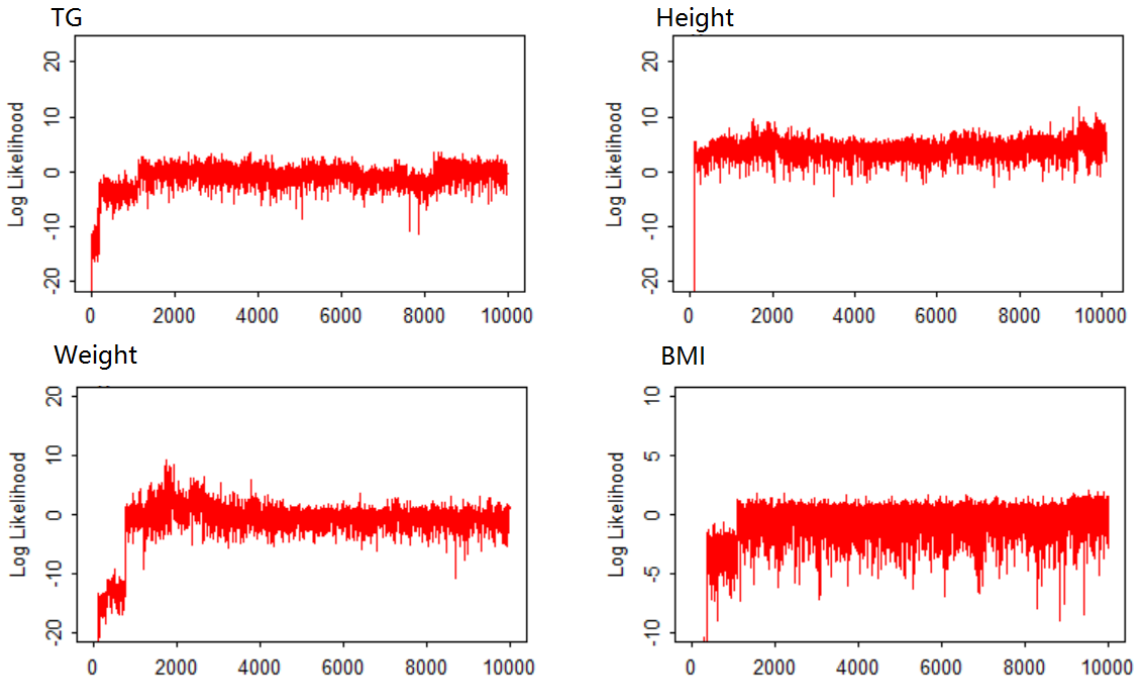


506

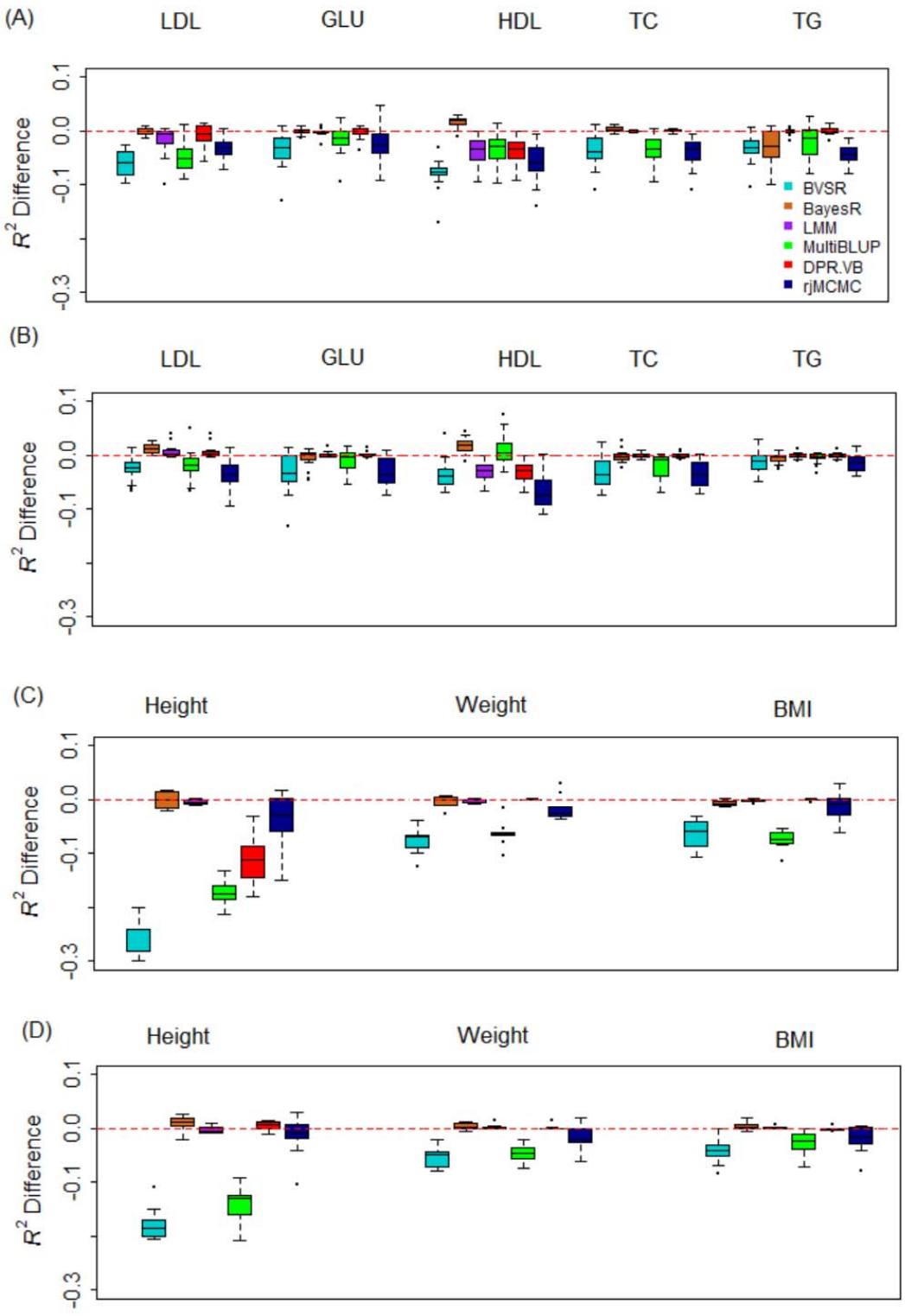
507 **Supplementary Figure 7. Comparison of prediction performance of several methods**
508 **with DPR.MCMC for twelve traits from three data sets.** Performance is measured by
509 MSE difference with respect to DPR.MCMC, where a positive value (i.e. values above
510 the red horizontal line) indicates worse performance than DPR.MCMC. Methods for
511 comparison include BVSR (cyan), BayesR (chocolate), LMM (purple), MultiBLUP
512 (green), DPR.VB (red), rjMCMC (black blue) and DPR.MCMC. The sample MSE
513 differences are obtained from 20 replicates of Monte Carlo cross validation for each trait.
514 For each box plot, the bottom and top of the box are the first and third quartiles, while the
515 ends of whiskers represent either the lowest datum within 1.5 interquartile range of the

lower quartile or the highest datum within 1.5 interquartile range of the upper quartile. For DPR.MCMC, the mean predictive MSE in the test set and the standard deviation are 0.246 (0.011) for MFP, 0.371 (0.019) for MY, 0.446 (0.028) for SCS, 0.170 (0.012) for GDD, 0.928 (0.029) for LDL, 0.954 (0.034) for GLU, 0.833 (0.063) for HDL, 0.970 (0.044) for TC, 0.960 (0.035) for TG, 0.519 (0.050) for height, 0.834 (0.065) for weight and 0.868 (0.074) for BMI. The SNP heritability estimates are 0.912 (0.007) for MFP, 0.810 (0.012) for MY, 0.801 (0.012) for SCS, 0.880 (0.013) for GDD, 0.397 (0.024) for LDL, 0.357 (0.036) for GLU, 0.418 (0.024) for HDL, 0.402 (0.036) for TC, 0.334 (0.034) for TG, 0.905 (0.013) for Height, 0.548 (0.022) for Weight and 0.483 (0.023) for BMI.


MFP
MY
SCS
GDD
LDL
GLU
HDL
TC
Log Likelihood



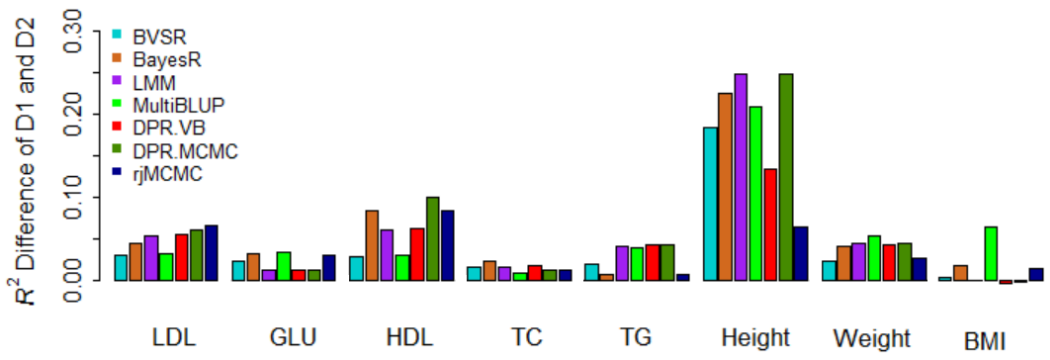
Supplementary Figure 8. Trace plots of the log posterior likelihood of DPR.MCMC in real data applications. For each of the twelve traits in the three GWAS data sets, we plot the log posterior likelihood versus the first 10,000 iterations (i.e. burn-in period) using the first cross-validation data. In each panel, the log posterior likelihood values were centered to have a median value of zero.



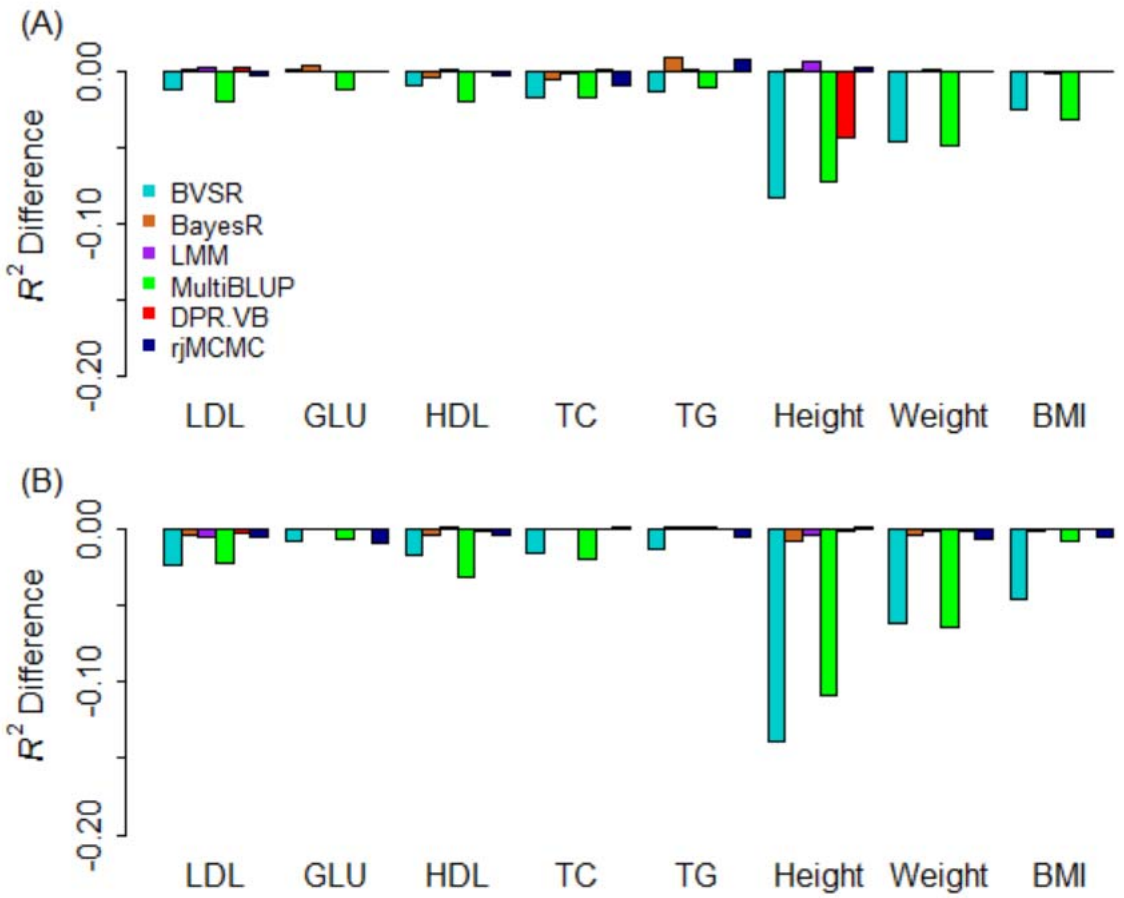
564

565 **Supplementary Figure 9. Comparison of prediction performance of several methods**
566 **with DPR.MCMC for eight traits in each of the two sub data sets of FHS.** The two
567 sub data sets D1 and D2 have the same sample size but different levels of relatedness

(individuals in D1 are more related to each other than those in D2). (A) The R^2 difference of five plasma traits (LDL, GLU, HDL, TC and TG) with respect to DPR.MCMC in the D1 and D2 sub data of FHS; (B) The R^2 difference of three anthropometric traits (Height, Weight and BMI) with respect to DPR.MCMC in the D1 and D2 sub data of FHS. For each box plot, the bottom and top of the box are the first and third quartiles, while the ends of whiskers represent either the lowest datum within 1.5 interquartile range of the lower quartile or the highest datum within 1.5 interquartile range of the upper quartile. FHS: Framingham heart study.



Supplementary Figure 10. Prediction performance of various methods are higher in a data with more related individuals (D1) than in a data with less related individuals (D2). The two data sets D1 and D2 from FHS have the same sample size but different levels of relatedness (individuals in D1 are more related to each other than those in D2). For each trait in the FHS data (x-axis), we first computed the median predictive R^2 across 20 replicates in D1 and D2 separately, and then contrast the difference between the two averaged predictive R^2 values in the two data sets (D1 minus D2; y-axis). Positive averaged predictive R^2 differences suggest that all methods have higher predictive performance in D1 versus D2. FHS: Framingham heart study.



Supplementary Figure 11. Comparison of prediction performance of several methods with DPR.MCMC using cross-validation between the two sub data sets of FHS. The two sub data sets D1 and D2 have the same sample size but different levels of relatedness (individuals in D1 are more related to each other than those in D2). (A) Predictive R^2 difference of different methods in D1 using parameters inferred in D2. For DPR.MCMC, the R^2 is 0.024 for LDL, 0.012 for GLU, 0.021 for HDL, 0.022 for TC, 0.016 for TG, 0.131 for Height, 0.061 for Weight and 0.041 for BMI. (B) Predictive R^2 difference of different methods in D2 using parameters inferred in D1; For DPR.MCMC, the R^2 is 0.043 for LDL, 0.009 for GLU, 0.033 for HDL, 0.021 for TC, 0.015 for TG, 0.226 for Height, 0.083 for Weight and 0.058 for BMI. FHS: Framingham heart study.

Supplementary Table 1. Sampling variation of R^2 measured by standard deviation across Monte Carlo cross validation replicates for various methods in simulations and real data analysis.

		BVSR	rjMCMC	BayesR	LMM	MultiBLUP	DPR	
							VB	MCMC
Simulations								
PVE = 0.2								
I		0.019	0.019	0.020	0.019	0.019	0.019	0.019
II		0.016	0.016	0.016	0.015	0.015	0.016	0.016
III	10	0.017	0.017	0.019	0.018	0.018	0.017	0.017
	100	0.018	0.018	0.018	0.019	0.019	0.018	0.018
	1,000	0.015	0.015	0.015	0.016	0.016	0.015	0.015
IV	normal	0.023	0.023	0.023	0.023	0.023	0.023	0.023
	t	0.016	0.016	0.016	0.015	0.015	0.016	0.016
	Laplace	0.017	0.017	0.017	0.017	0.017	0.017	0.017
PVE = 0.5								
I		0.031	0.030	0.030	0.030	0.030	0.031	0.031
II		0.024	0.028	0.026	0.028	0.028	0.027	0.031
III	10	0.029	0.026	0.027	0.027	0.027	0.031	0.028
	100	0.031	0.031	0.031	0.030	0.030	0.031	0.031
	1,000	0.031	0.031	0.031	0.030	0.030	0.031	0.031
IV	normal	0.030	0.030	0.031	0.030	0.031	0.030	0.030
	t	0.025	0.025	0.025	0.027	0.026	0.025	0.025
	Laplace	0.023	0.023	0.023	0.024	0.024	0.024	0.024
PVE = 0.8								
I		0.027	0.029	0.029	0.028	0.028	0.029	0.029
II		0.028	0.022	0.022	0.022	0.022	0.022	0.024
III	10	0.022	0.024	0.022	0.023	0.023	0.024	0.024
	100	0.032	0.028	0.027	0.026	0.026	0.028	0.027
	1,000	0.035	0.030	0.030	0.030	0.030	0.030	0.030
IV	normal	0.030	0.028	0.028	0.028	0.028	0.028	0.028
	t	0.027	0.027	0.026	0.027	0.027	0.027	0.027
	Laplace	0.024	0.022	0.022	0.022	0.022	0.022	0.022
Real data								
Cattle								
	MFP	0.013	0.012	0.011	0.013	0.030	0.018	0.011
	MY	0.015	0.013	0.012	0.013	0.013	0.014	0.012
	SCS	0.019	0.020	0.018	0.018	0.016	0.022	0.017
Maize								
	GDD	0.013	0.011	0.012	0.010	0.014	0.013	0.012
FHS								
	LDL	0.013	0.013	0.032	0.014	0.033	0.014	0.012
	GLU	0.010	0.010	0.022	0.015	0.022	0.016	0.012
	HDL	0.010	0.021	0.029	0.015	0.067	0.018	0.019
	TC	0.011	0.014	0.019	0.009	0.020	0.016	0.015
	TG	0.008	0.014	0.018	0.020	0.022	0.011	0.014
	Height	0.032	0.047	0.051	0.045	0.048	0.050	0.050
	Weight	0.037	0.042	0.040	0.029	0.040	0.042	0.040
	BMI	0.034	0.038	0.036	0.035	0.036	0.041	0.039

Supplementary Table 2. Significant genes identified by DPR.MCMC for different diseases in the PrediXcan gene set analysis of WTCCC.

Disease	Gene	Chr	TSS	z score	p value	#SNPs	h^2	References
T1D	*LINC00240*[M,V]	6	26,988,232	5.73	9.78E-09	277	0.255	28
T1D	*ZNF165*[M,V]	6	28,048,753	7.40	1.40E-13	396	0.231	28
T1D	*ZNF192*[M,V]	6	28,109,716	6.80	1.04E-11	387	0.041	28
T1D	*TRIM3*[H,V]	6	30,080,883	-6.77	1.30E-11	13	0.089	28,29
T1D	*HCG18*[H,V]	6	30,294,927	-5.42	5.85E-08	9	0.468	29-33
T1D	*IER3*[H,V]	6	30,712,331	-7.07	1.60E-12	35	0.405	29-33
T1D	*DDR1*[H,V]	6	30,844,198	-7.31	2.76E-13	24	0.217	29-33
T1D	*VARS2*[H,V]	6	30,876,019	-5.05	4.34E-07	16	0.195	29-33
T1D	*MUC22*[H,V]	6	30,978,251	5.85	5.05E-09	148	0.155	29-33
T1D	*HCG22*[H,V]	6	31,021,227	-4.54	5.55E-06	177	0.719	29-33
T1D	*HLA-B*[H,V]	6	31,324,965	4.74	2.12E-06	153	0.579	29-33
T1D	*MICA*[H,V]	6	31,367,561	4.81	1.50E-06	114	0.157	29-33
T1D	*MICB*[H,V]	6	31,462,658	4.45	8.59E-06	66	0.620	29-33
T1D	*LST1*[H,V]	6	31,553,901	14.49	1.46E-47	42	0.377	29-33
T1D	*AGPAT1*[H,V]	6	32,145,873	-9.50	2.04E-21	13	0.046	29-33
T1D	*HLA-DRB5*[H,V]	6	32,498,064	-5.04	4.70E-07	28	0.741	29-33
T1D	*HLA-DQA2*[G]	6	32,709,119	18.85	2.99E-79	103	0.709	33
T1D	*HLA-DQB2*[H,V]	6	32,731,311	10.78	4.15E-27	119	0.778	33
T1D	*TAP2*[H,V]	6	32,806,599	-4.43	9.45E-06	111	0.815	33
T1D	*PSMB9*[H,V]	6	32,811,913	4.71	2.44E-06	120	0.205	33
T1D	*TAP1*[H,V]	6	32,821,755	8.60	7.70E-18	113	0.066	33
T1D	*HLA-DOA*[H,V]	6	32,977,389	-7.36	1.88E-13	55	0.152	33
T1D	*HLA-DPA1*[H,V]	6	33,048,552	6.80	1.04E-11	73	0.423	33,34
T1D	*HSD17B8*[H,V]	6	33,172,419	7.99	1.40E-15	46	0.194	33,34
T1D	*RPS26*[G]	12	56,435,637	5.93	2.97E-09	74	0.805	31
CD	*POU5F1*[H,V]	6	31,148,508	4.23	2.35E-05	260	0.526	31,35-39
CD	*LINC00481*[H,V]	6	31,169,695	4.47	7.70E-06	256	0.281	31,35-39
CD	*PTGER4*[G]	5	40,679,600	5.31	1.11E-07	292	0.182	40
CD	*AC091132.3*[V]	17	43,595,264	4.48	7.40E-06	24	0.557	35,37,41
CD	*PTPN2*[G]	18	12,884,337	-5.01	5.58E-07	194	0.260	31,37,40,41
CD	*STMN3*[V]	20	62,284,780	-4.43	9.38E-06	96	0.277	37
RA	*PANK4*[V]	1	2,458,039	4.39	1.13E-05	64	0.126	42-44
RA	*HLA-G*[G]	6	29,794,744	4.54	5.57E-06	64	0.459	43,45-57
RA	*TRIM26*[V]	6	30,181,204	-5.85	4.80E-09	12	0.044	45
RA	*IER3*[V]	6	30,712,331	-5.23	1.72E-07	35	0.405	43,45-57
RA	*HLA-DRB5*[V]	6	32,498,064	-6.84	8.11E-12	28	0.741	43,45-57

RA	*HLA-DQA2*[G]	6	32,709,119	9.51	1.82E-21	103	0.709	52,58
RA	*HLA-DQB2*[V]	6	32,731,311	9.38	6.88E-21	119	0.778	43,45-57

The table also lists the disease name, gene id, chromosome number, transcription start site (TSS), association strength (z score, p value), the number of SNPs in each gene set test, estimated SNP heritability (h^2, from GEMMA), and references that support the identified association. T1D: type 1 diabetes, CD: Crohn's disease, RA: rheumatoid arthritis. H indicates Human leukocyte antigen (HLA) region genes on chromosome 6, M indicates major histocompatibility complex (MHC) region, G indicates genes previously identified to be associated with diseases in the NHGRI GWAS catalog, V indicates the vicinity of a reported gene. h^2 is the estimator of heritability using linear mixed models in GEMMA.

Supplementary References

1. Zhou, X., Carbonetto, P., & Stephens, M. Polygenic modeling with Bayesian sparse linear mixed models. *PLoS Genet.* **9,** e1003264 (2013).
2. Yang, J. *et al.* Common SNPs explain a large proportion of the heritability for human height. *Nat. Genet.* **42,** 565-569 (2010).
3. Moser, G. *et al.* Simultaneous Discovery, Estimation and Prediction Analysis of Complex Traits Using a Bayesian Mixture Model. *PLoS Genet.* **11,** e1004969 (2015).
4. Robert, C., & Casella, G. *Monte Carlo statistical methods* (Second ed.). New York: Springer (2002).
5. Gelman, A. Parameterization and Bayesian Modeling. *J. Am. Stat. Assoc.* **99,** 537-545 (2004).
6. Visscher, P. M., Hill, W. G., & Wray, N. R. Heritability in the genomics era--concepts and misconceptions. *Nat. Rev. Genet.* **9,** 255-266 (2008).
7. de los Campos, G., Sorensen, D., & Gianola, D. Genomic heritability: what is it? *PLoS Genet.* **11,** e1005048 (2015).
8. Zhou, X., & Stephens, M. Genome-wide efficient mixed-model analysis for association studies. *Nat. Genet.* **44,** 821-824 (2012).
9. Lippert, C. *et al.* FaST linear mixed models for genome-wide association studies. *Nat. Methods* **8,** 833-835 (2011).
10. Levine, R. A., & Casella, G. Optimizing random scan Gibbs samplers. *J. Multivariate Anal.* **97,** 2071-2100 (2006).
11. Levine, R. A., Yu, Z., Hanley, W. G., & Nitao, J. J. Implementing random scan Gibbs samplers. *Comput Stat* **20,** 177-196 (2005).
12. Blei, D. M., & Jordan, M. I. Variational inference for Dirichlet process mixtures. *Bayesian. Anal.* **1,** 121-143 (2006).
13. Ishwaran, H., & James, L. F. Approximate Dirichlet Process Computing in Finite Normal Mixtures. *J. Comput. Graph. Statist.* **11,** 508-532 (2002).
14. Ishwaran, H., & James, L. F. Gibbs sampling methods for stick-breaking priors. *J. Am. Stat. Assoc.* **96,** (2001).
15. Gelman, A. *et al. Bayesian Data Analysis* (Third ed.). New York: Chapman & Hall/CRC (2013).
16. Spiegelhalter, D. J., Best, N. G., Carlin, B. P., & Van Der Linde, A. Bayesian measures of model complexity and fit. *J. R. Stat. Soc. Ser. B.* **64,** 583-639 (2002).
17. Gelman, A., Hwang, J., & Vehtari, A. Understanding predictive information criteria for Bayesian models. *Stat. Comput.* **24,** 997-1016 (2014).
18. Brooks, S. Markov chain Monte Carlo method and its application. *Journal of the royal statistical society: series D (the Statistician)* **47,** 69-100 (1998).
19. Hastie, T., Tibshirani, R., & Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction*. New York, NY: Springer (2009).
20. Bishop, C. M. *Pattern recognition and machine learning*. New York: Springer (2006).
21. Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., & Saul, L. K. An introduction to variational methods for graphical models. *Mach. Learn.* **37,** 183-233 (1999).
22. Grimmer, J. An Introduction to Bayesian Inference via Variational Approximations. *Pol. Anal.* **19,** 32-47 (2011).

23. Ormerod, J. T., & Wand, M. Explaining variational approximations. *Am. Stat.* **64,** 140-153 (2010).
24. Pham, T. H., Ormerod, J. T., & Wand, M. P. Mean field variational Bayesian inference for nonparametric regression with measurement error. *Comput. Stat. Data Anal.* **68,** 375-387 (2013).
25. Wand, M. P., Ormerod, J. T., Padoan, S. A., & Fuhrwirth, R. Mean field variational Bayes for elaborate distributions. *Bayesian. Anal.* **6,** 847-900 (2011).
26. Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. Variational inference: A review for statisticians. *J. Am. Stat. Assoc. (in press), Preprint at https://arxiv.org/abs/1601.00670* (2017).
27. Wang, C., & Blei, D. M. Variational inference in nonconjugate models. *J. Mach. Learn. Res.* **14,** 1005-1031 (2013).
28. DIAbetes Genetics Replication And Meta-analysis (DIAGRAM) Consortium *et al.* Genome-wide trans-ancestry meta-analysis provides insight into the genetic architecture of type 2 diabetes susceptibility. *Nat. Genet.* **46,** 234-244 (2014).
29. Barrett, J. C. *et al.* Genome-wide association study and meta-analysis find that over 40 loci affect risk of type 1 diabetes. *Nat. Genet.* **41,** 703-707 (2009).
30. Cooper, J. D. *et al.* Meta-analysis of genome-wide association study data identifies additional type 1 diabetes risk loci. *Nat. Genet.* **40,** 1399-1401 (2008).
31. The Wellcome Trust Case Control Consortium. Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature* **447,** 661-678 (2007).
32. Hakonarson, H. *et al.* A genome-wide association study identifies KIAA0350 as a type 1 diabetes gene. *Nature* **448,** 591-594 (2007).
33. Perry, J. R. *et al.* Stratifying type 2 diabetes cases by BMI identifies genetic risk variants in LAMA1 and enrichment for risk variants in lean compared to obese cases. *PLoS Genet.* **8,** e1002741 (2012).
34. Lin, H. *et al.* Novel susceptibility genes associated with diabetic cataract in a Taiwanese population. *Ophthalmic Genet.* **34,** 35-42 (2013).
35. Yamazaki, K. *et al.* A genome-wide association study identifies 2 susceptibility loci for Crohn's disease in a Japanese population. *Gastroenterology* **144,** 781-788 (2013).
36. Jostins, L. *et al.* Host-microbe interactions have shaped the genetic architecture of inflammatory bowel disease. *Nature* **491,** 119-124 (2012).
37. Franke, A. *et al.* Genome-wide meta-analysis increases to 71 the number of confirmed Crohn's disease susceptibility loci. *Nat. Genet.* **42,** 1118-1125 (2010).
38. Julià, A. *et al.* A genome-wide association study on a southern European population identifies a new Crohn's disease susceptibility locus at RBX1-EP300. *Gut* **62,** 1440-1445 (2013).
39. Yang, S. K. *et al.* Genome-wide association study of Crohn's disease in Koreans revealed three new susceptibility loci and common attributes of genetic susceptibility across ethnic populations. *Gut* **63,** 80-87 (2014).
40. Parkes, M. *et al.* Sequence variants in the autophagy gene IRGM and multiple other replicating loci contribute to Crohn's disease susceptibility. *Nat. Genet.* **39,** 830-832 (2007).

41. Barrett, J. C. *et al.* Genome-wide association defines more than 30 distinct susceptibility loci for Crohn's disease. *Nat. Genet.* **40,** 955-962 (2008).
42. Orozco, G. *et al.* Novel Rheumatoid Arthritis Susceptibility Locus at 22q12 Identified in an Extended UK Genome-Wide Association Study. *Arthritis Rheumatol.* **66,** 24-30 (2014).
43. Stahl, E. A. *et al.* Genome-wide association study meta-analysis identifies seven new rheumatoid arthritis risk loci. *Nat. Genet.* **42,** 508-514 (2010).
44. Raychaudhuri, S. *et al.* Common variants at CD40 and other loci confer risk of rheumatoid arthritis. *Nat. Genet.* **40,** 1216-1223 (2008).
45. Eleftherohorinou, H., Hoggart, C. J., Wright, V. J., Levin, M., & Coin, L. J. Pathway-driven gene stability selection of two rheumatoid arthritis GWAS identifies and validates new susceptibility genes in receptor mediated signalling pathways. *Hum. Mol. Genet.* **20,** 3494-3506 (2011).
46. Hüffmeier, U. *et al.* Common variants at TRAF3IP2 are associated with susceptibility to psoriatic arthritis and psoriasis. *Nat. Genet.* **42,** 996-999 (2010).
47. Bossini-Castillo, L. *et al.* A genome-wide association study of rheumatoid arthritis without antibodies against citrullinated peptides. *Ann. Rheum. Dis.* annrheumdis-2013-204591 (2014).
48. Hu, H.-J. *et al.* Common variants at the promoter region of the APOM confer a risk of rheumatoid arthritis. *Exp. Mol. Med.* **43,** 613-621 (2011).
49. Terao, C. *et al.* The human AIRE gene at chromosome 21q22 is a genetic determinant for the predisposition to rheumatoid arthritis in Japanese population. *Hum. Mol. Genet.* **20,** 2680-2685 (2011).
50. Orozco, G. *et al.* Novel Rheumatoid Arthritis Susceptibility Locus at 22q12 Identified in an Extended UK Genome‐Wide Association Study. *Arthritis Rheumatol.* **66,** 24-30 (2014).
51. Behrens, E. M. *et al.* Association of the TRAF1–C5 locus on chromosome 9 with juvenile idiopathic arthritis. *Arthritis Rheum.* **58,** 2206-2207 (2008).
52. Nakajima, M. *et al.* New sequence variants in HLA class II/III region associated with susceptibility to knee osteoarthritis identified by genome-wide association study. *PLoS ONE* **5,** e9723 (2010).
53. Okada, Y. *et al.* Genetics of rheumatoid arthritis contributes to biology and drug discovery. *Nature* **506,** 376-381 (2014).
54. Jiang, L. *et al.* Novel risk loci for rheumatoid arthritis in Han Chinese and congruence with risk variants in Europeans. *Arthritis Rheumatol.* **66,** 1121-1132 (2014).
55. Padyukov, L. *et al.* A genome-wide association study suggests contrasting associations in ACPA-positive versus ACPA-negative rheumatoid arthritis. *Ann. Rheum. Dis.* (2010).
56. Plenge, R. M. *et al.* TRAF1–C5 as a risk locus for rheumatoid arthritis—a genomewide study. *N. Engl. J. Med.* **357,** 1199-1209 (2007).
57. Freudenberg, J. *et al.* Genome-wide association study of rheumatoid arthritis in Koreans: Population-specific loci as well as overlap with European susceptibility loci. *Arthritis Rheum.* **63,** 884-893 (2011).

760 58. Julia, A. *et al.* Genome‐wide association study of rheumatoid arthritis in the
761 Spanish population: KLF12 as a risk locus for rheumatoid arthritis susceptibility.
762 *Arthritis Rheum.* **58,** 2275-2286 (2008).
763
764