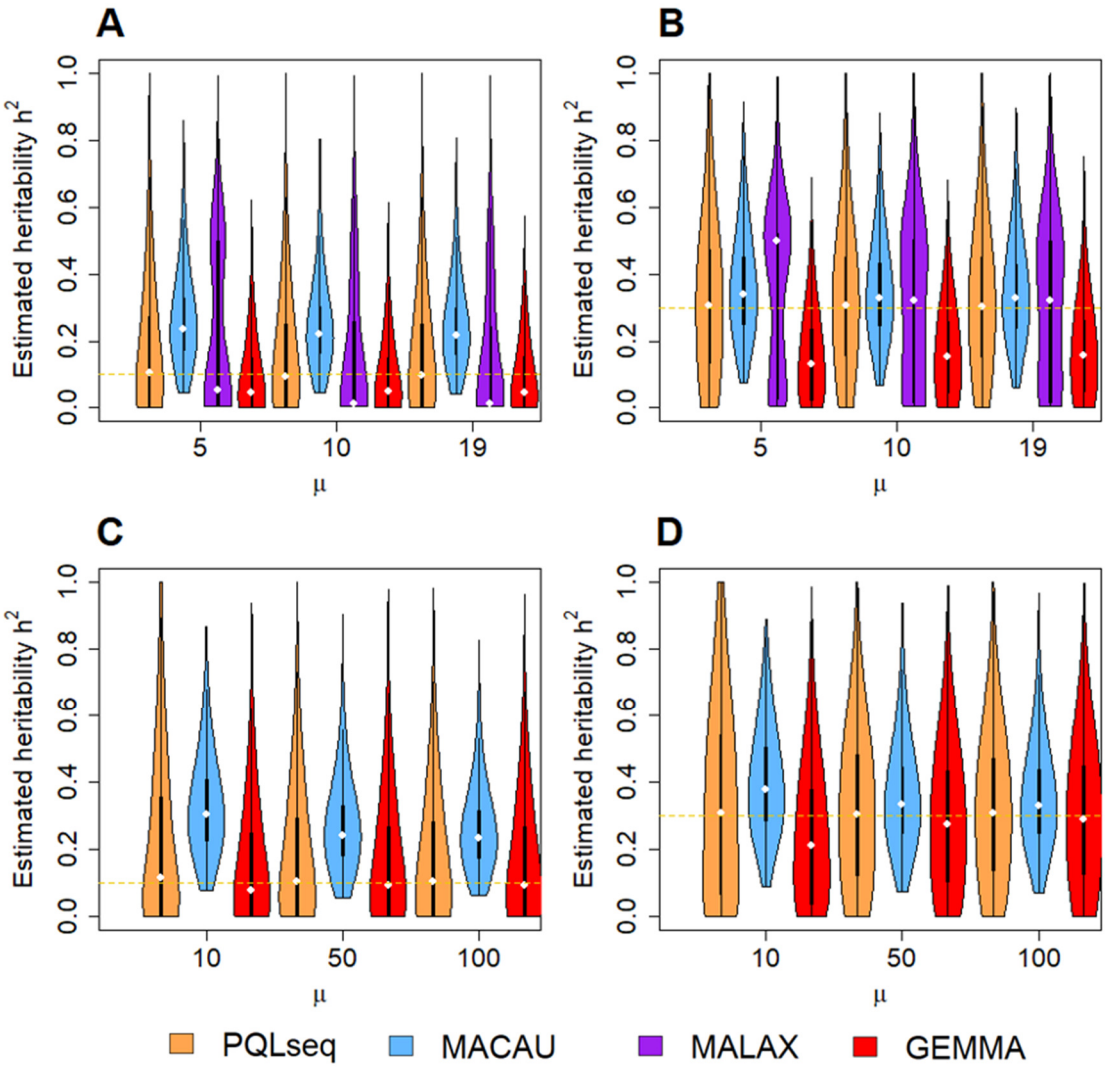
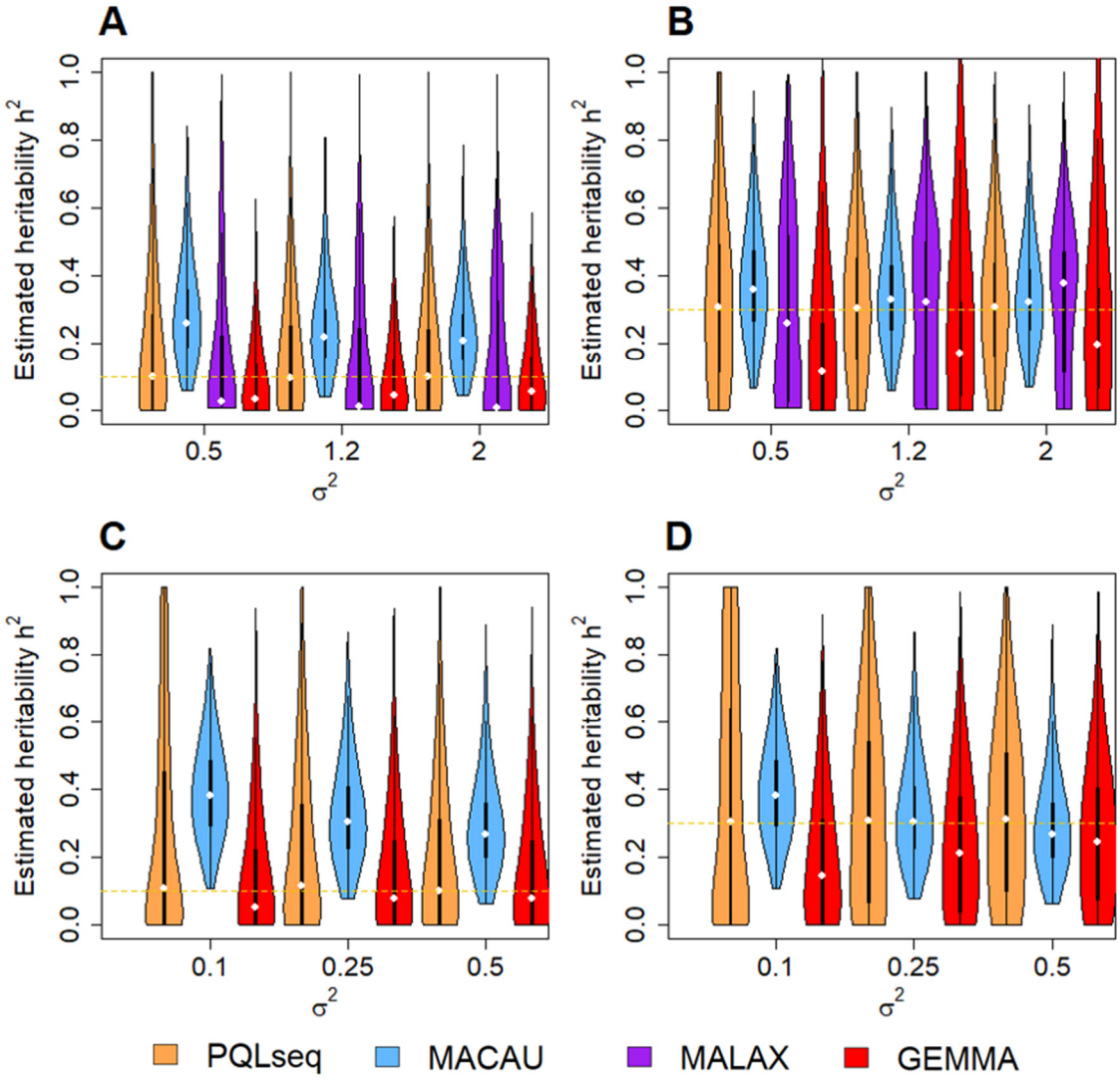
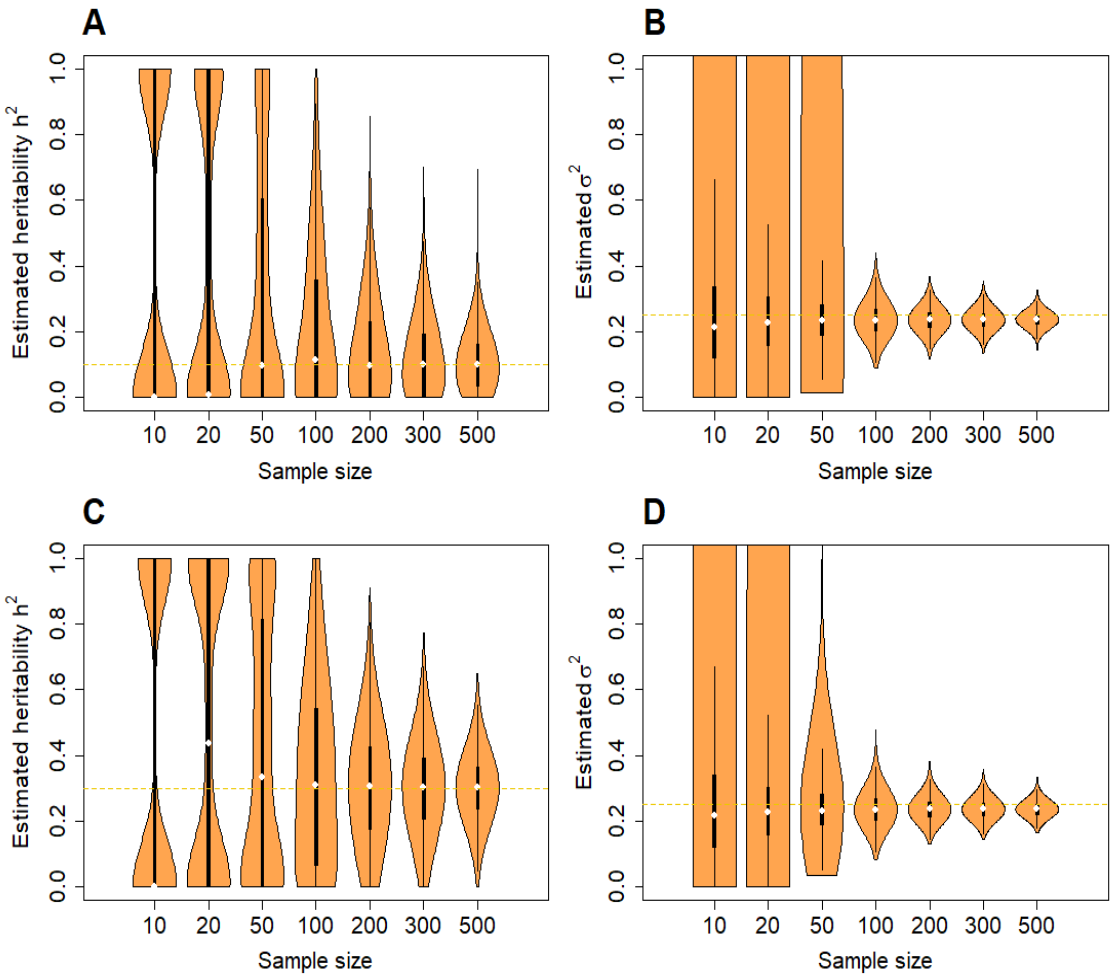
Supplementary Figure 1. PQLseq produces accurate heritability estimates regardless of the mean read count μ in simulated BSseq and RNAseq data. The heritability estimations are from PQLseq (orange), MACAU (blue), MALAX (purple) and GEMMA (red) across a range of μ (5, 10, 19 in BSseq simulations;10, 50 and 100 in RNAseq simulations). (**A**): $h^2 = 0.1$ and $\sigma^2 = 1.2$ in BSseq; (**B**): $h^2 = 0.3$ and $\sigma^2 = 1.2$ in BSeq; (**C**): $h^2 = 0.1$ and $\sigma^2 = 0.25$ in RNAseq; (**D**): $h^2 = 0.3$ and $\sigma^2 = 0.25$ in RNAseq. The other parameter setting is $n = 100$. The horizontal orange dashed line represents the true heritability.



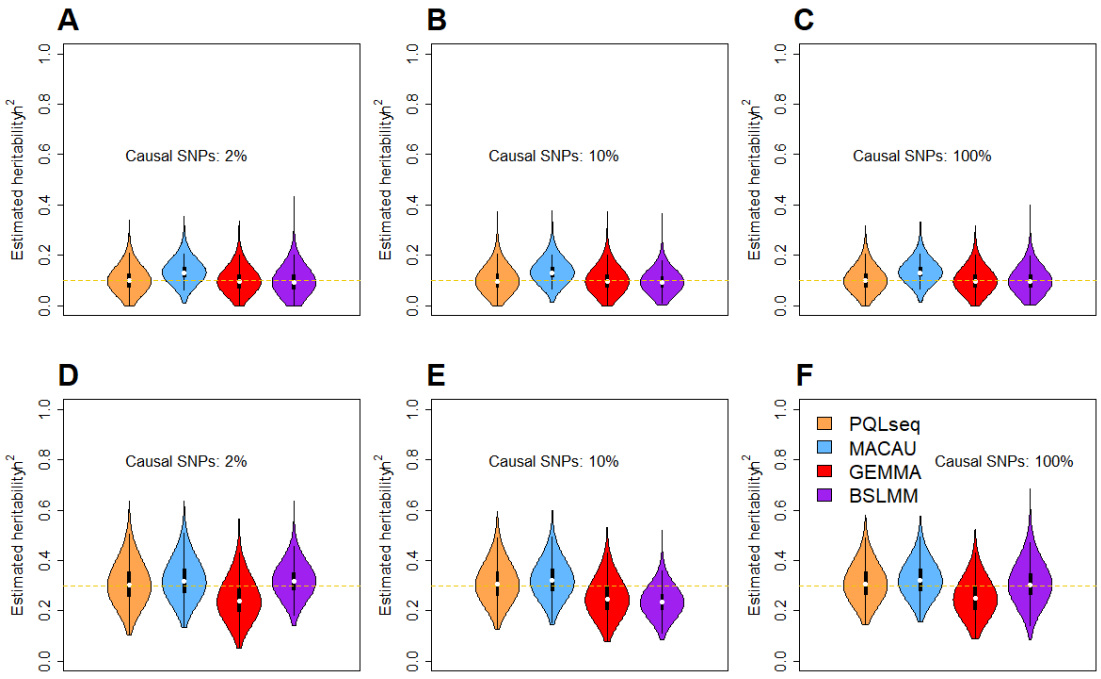
Supplementary Figure 2. PQLseq produces accurate heritability estimates regardless of the over-dispersion parameter σ^2 in simulated BSseq and RNAseq data. The heritability estimations are from PQLseq (orange), MACAU (blue), MALAX (purple), and GEMMA (red) across a range of σ^2 (0.5, 1.2, and 2 in BSseq simulations; 0.1, 0.25 and 0.5 in RNAseq simulations). (**A**): $h^2 = 0.1$ and $\mu = 19$ in BSseq; (**B**): $h^2 = 0.3$ and $\mu = 19$ in BSeq; (**C**): $h^2 = 0.1$ and $\mu = 10$ in RNAseq; (**D**): $h^2 = 0.3$ and $\mu = 10$ in RNAseq. The other parameter setting is $n = 100$. The horizontal orange dashed line represents the true heritability.



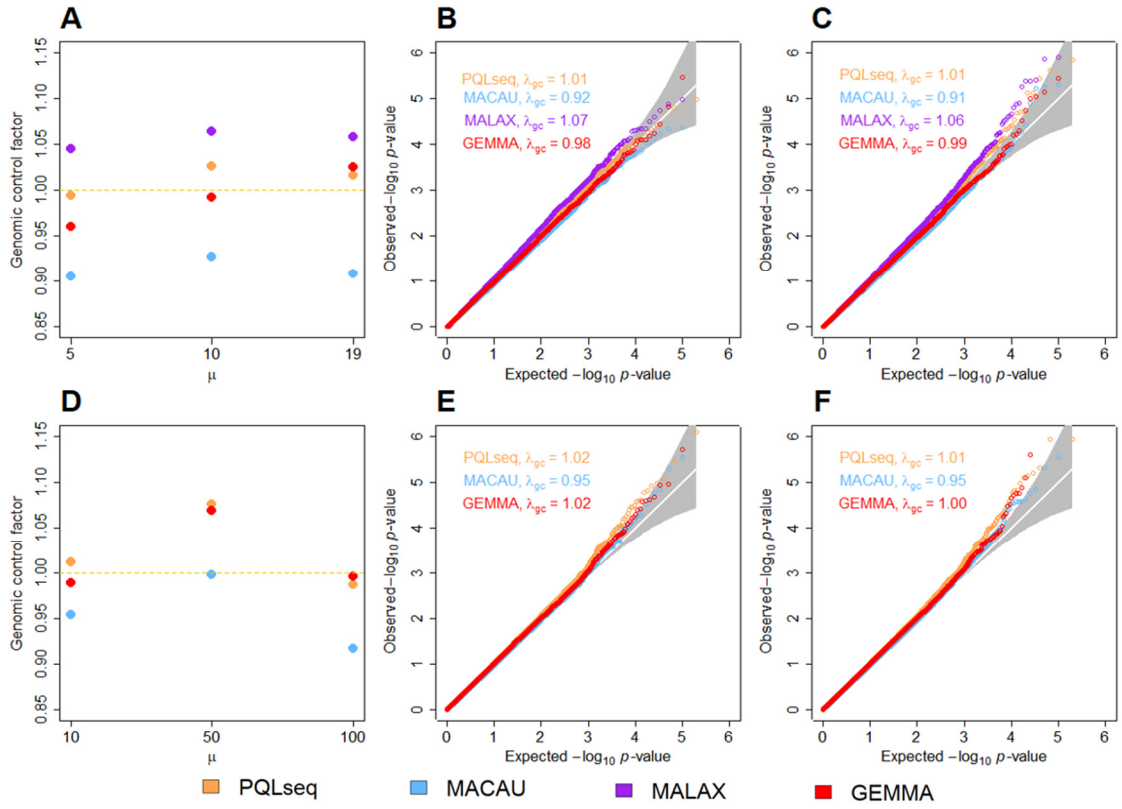
Supplementary Figure 3. Estimation of heritability h^2 and dispersion parameter σ^2 by PQLseq under the null simulations with varying sample sizes. The results are based on RNAseq based simulations with varying sample sizes (n = 10 to 500). **A** and **B**: heritability estimates (**A**) and dispersion parameter estimates (**B**) versus sample size under the setting of h^2 = 0.1 and σ^2 = 0.25; **C** and **D**: heritability estimates (**C**) and dispersion parameter estimates (**D**) versus sample size under the setting of h^2 = 0.3 and σ^2 = 0.25.



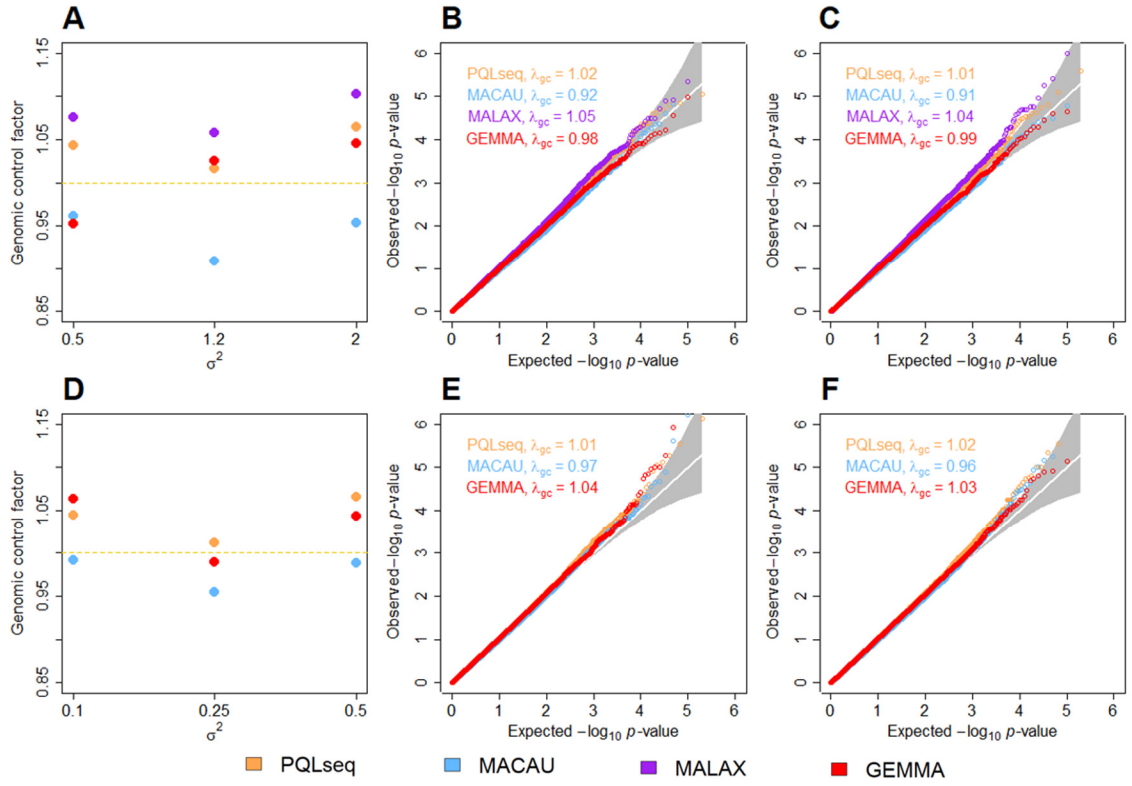
Supplementary Figure 4. PQLseq produces accurate heritability estimates under different genetic architectures. The SNP heritability estimations are from PQLseq (orange), MACAU (blue), GEMMA (red), and BSLMM (purple) under the null simulations for RNAseq based simulation. Parameters used include σ^2 = 0.25, PVE = 0.25, n = 465, together with h^2 = 0.1 and 2% causal SNPs (**A**); h^2 = 0.1 and 10% causal SNPs (**B**); h^2 = 0.1 and 100% causal SNPs (**C**); h^2 = 0.3 and 2% causal SNPs (**D**); h^2 = 0.3 and 10% causal SNPs (**E**); h^2 = 0.3 and 100% causal SNPs (**F**).



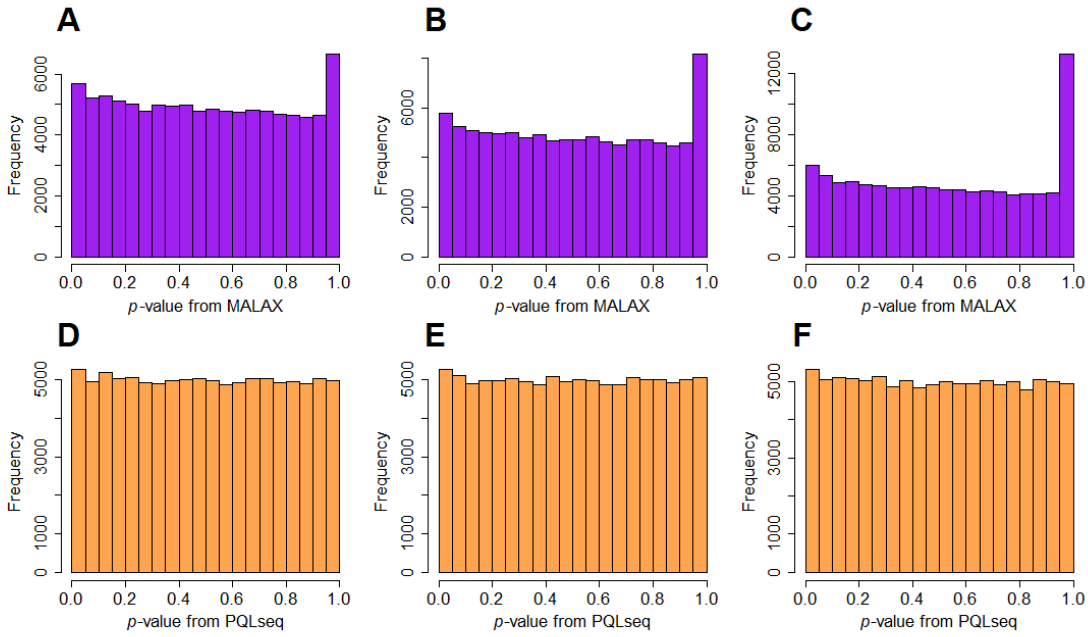
Supplementary Figure 5. PQLseq produces calibrated *p*-values regardless of the mean read count μ. Genomic control factors are from PQLseq (orange), MACAU (blue), MALAX (purple), and GEMMA (red) across a range of gene expression level μ under the null simulations are shown for BSseq based simulation (**A**) or RNAseq based simulation (**D**). Parameters used include σ^2 = 1.2 and n = 100 for BSseq based simulations and σ^2 = 0.25 and n = 100 for RNAseq simulations. QQ-plots further compare the expected and observed *p*-values (aggregates the results from 10 simulated datasets) distributions generated from different methods under the null for μ = 5 (**B**) and μ = 10 (**C**) in BSseq based simulations, and for μ = 50 (**E**) and μ = 100 (**F**) in RNAseq based simulations. λ_{gc} is the genomic control factor.



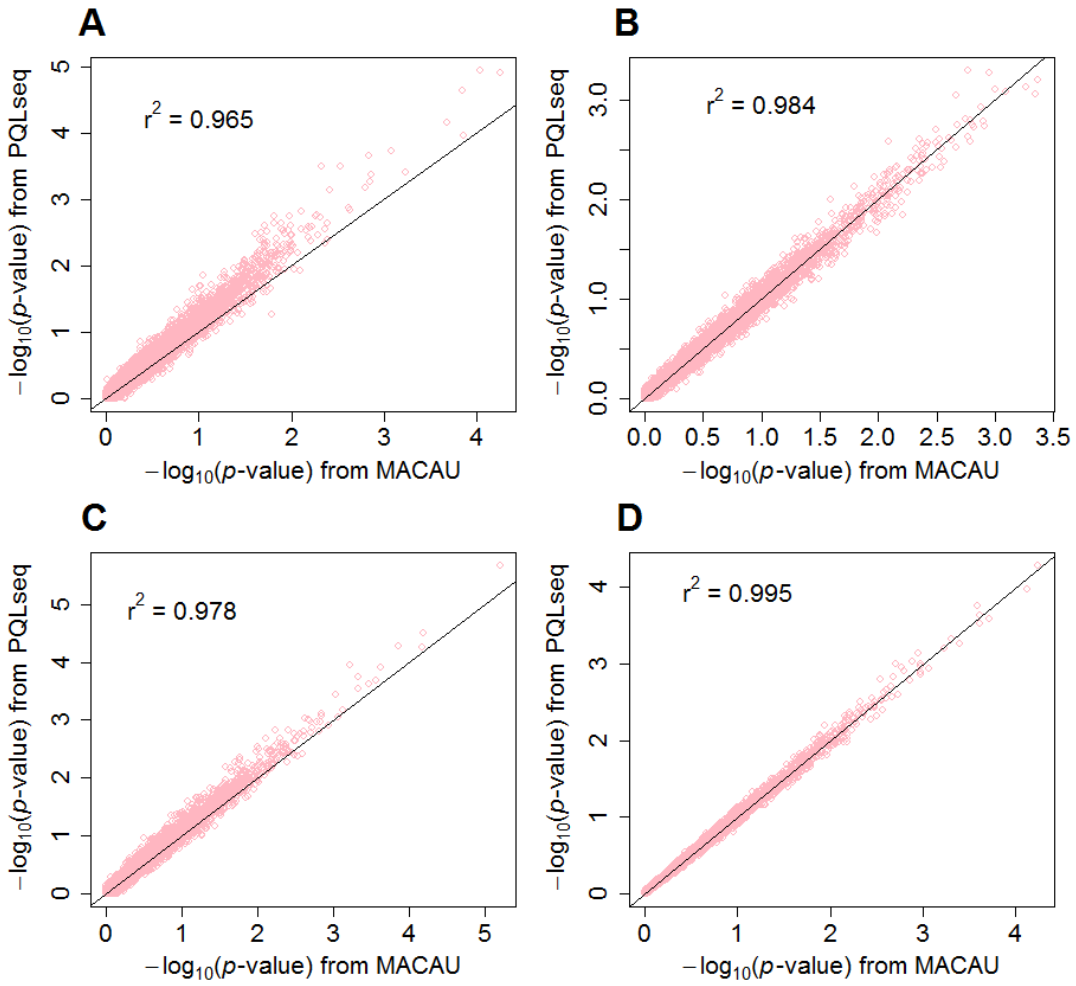
Supplementary Figure 6. PQLseq produces calibrated *p*-values regardless of the over-dispersion parameter σ^2. Genomic control factors from PQLseq (orange), MACAU (blue), MALAX (purple), and GEMMA (red) across a range of overdispersion paratemer σ^2 under the null simulations are shown for BSseq based simulation (**A**) or RNAseq based simulation (**D**). Parameters used include μ = 19 and n = 100 for BSseq based simulations and μ = 10 and n = 100 for RNAseq simulations. QQ-plots further compare the expected and observed *p*-values (aggregates the results from 10 simulated datasets) distributions generated from different methods under the null for σ^2 = 0.5 (**B**) and σ^2 = 2 (**C**) in BSseq based simulations, and for σ^2 = 0.1 (**E**) and σ^2 = 0.5 (**F**) in RNAseq based simulations. λ_{gc} is the genomic control factor.



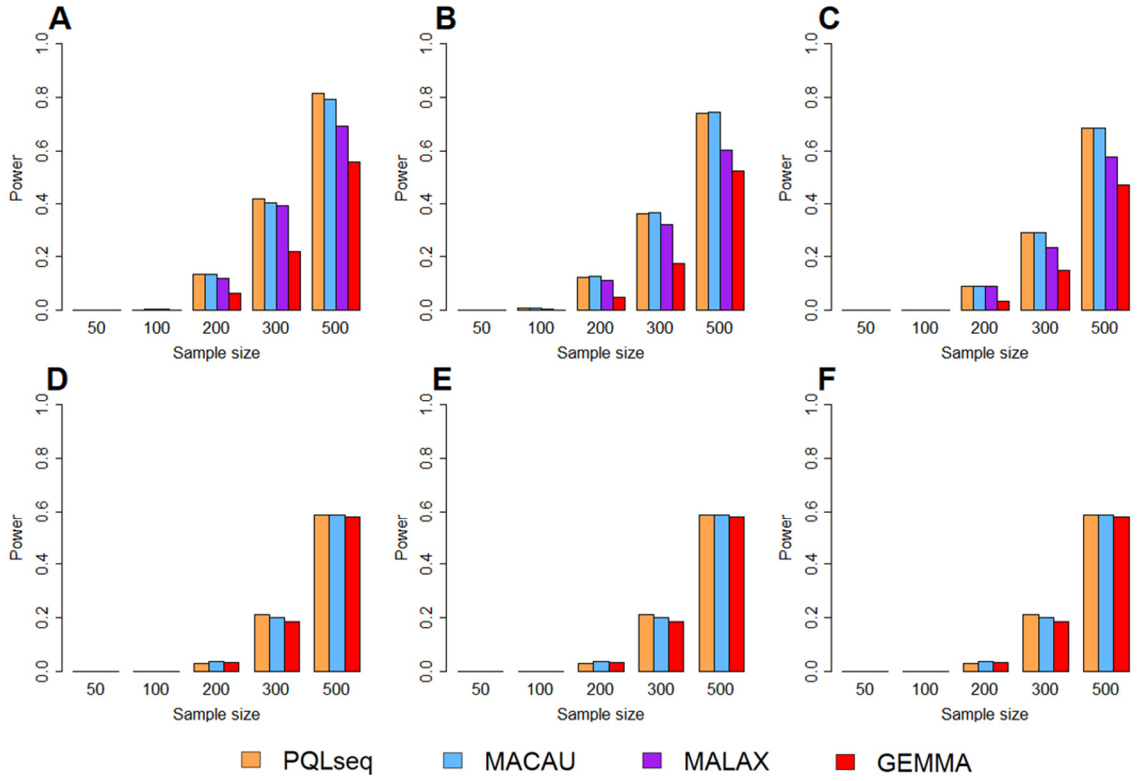
Supplementary Figure 7. *p*-values from MALAX are enriched near one in BSseq based simulations. Histograms of *p*-values by MALAX (purple; **A**, **B**, **C**) or *p*-values by by PQLseq (orange; **D**, **E**, **F**) aggregated from 10 null simulation replicates in BSseq based simulations. The results are based on different samples sizes: n = 200 (**A** and **D**), n = 300 (**B** and **E**), or n = 500 (**C** and **F**). The other parameters are $h^2 = 0.1$, $\sigma^2 = 2$, and μ = 19.



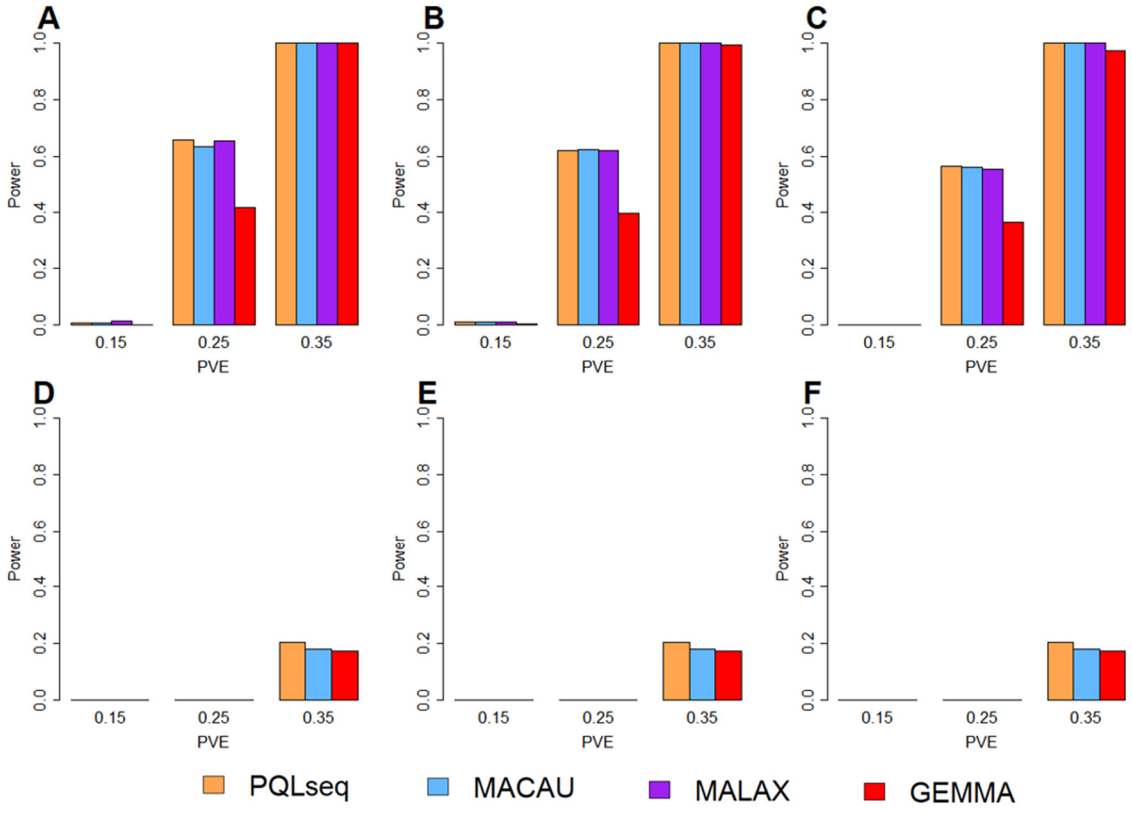
Supplementary Figure 8. PQLseq versus MACAU in simulated BSseq and RNAseq data. The *p*-values from the two methods are highly correlated in BSseq simulations with n = 50 (**A**) and n = 500 (**B**). The *p*-values from the both methods are highly correlated in RNAseq simulations with n = 50 (**C**) and n = 500 (**D**).



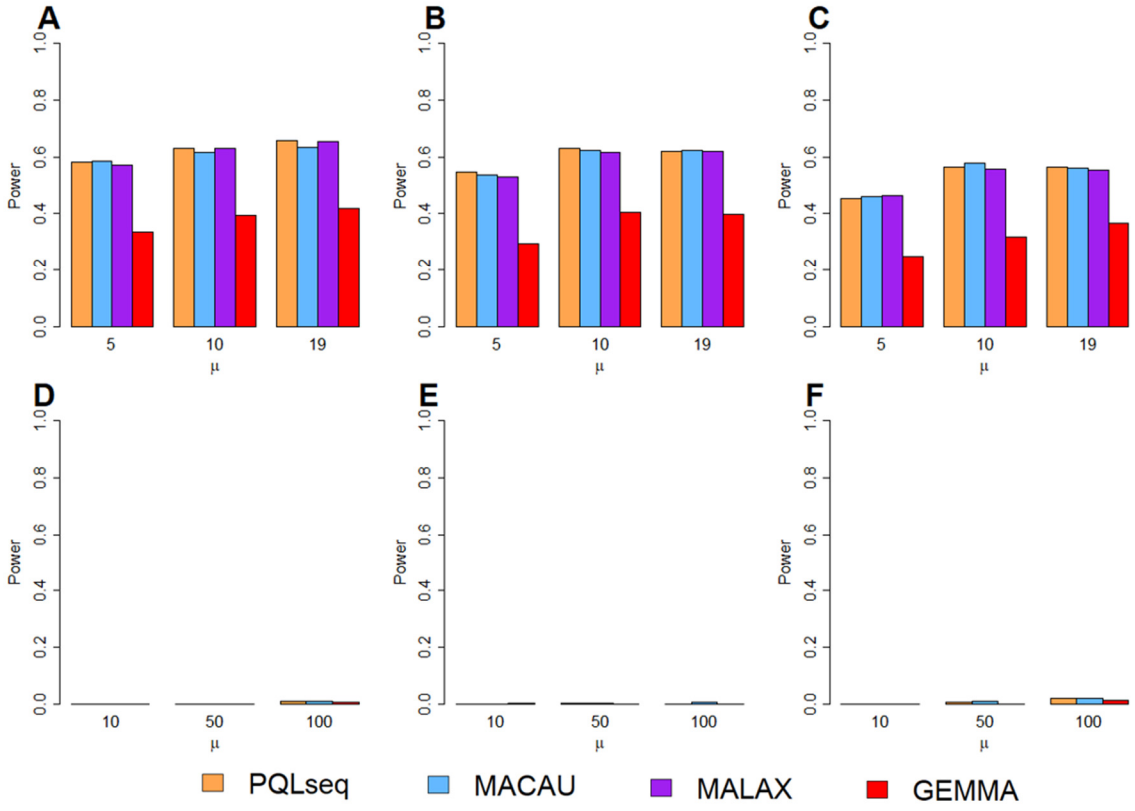
Supplementary Figure 9. PQLseq exhibits similar power as MACAU in BSseq and RNAseq based power simulations across a range of sample sizes and heritability values. The power results are obtained for PQLseq (orange), MACAU (blue), MALAX (purple) and GEMMA (red) based on 5% FDR in both BSseq based simulations (**A**, **B**, **C**) and RNAseq based simulations (**D**, **E**, **F**). Results are shown under different heritability values: h^2 = 0 (**A** and **D**), h^2 = 0.1 (**B** and **E**), or h^2=0.3 (**C** and **F**). The other parameter settings in the simulations are μ = 19, PVE = 0.15 and σ^2 = 1.2 for BSseq simulations; μ = 10, PVE = 0.25 and σ^2 = 0.25 for RNAseq simulations.



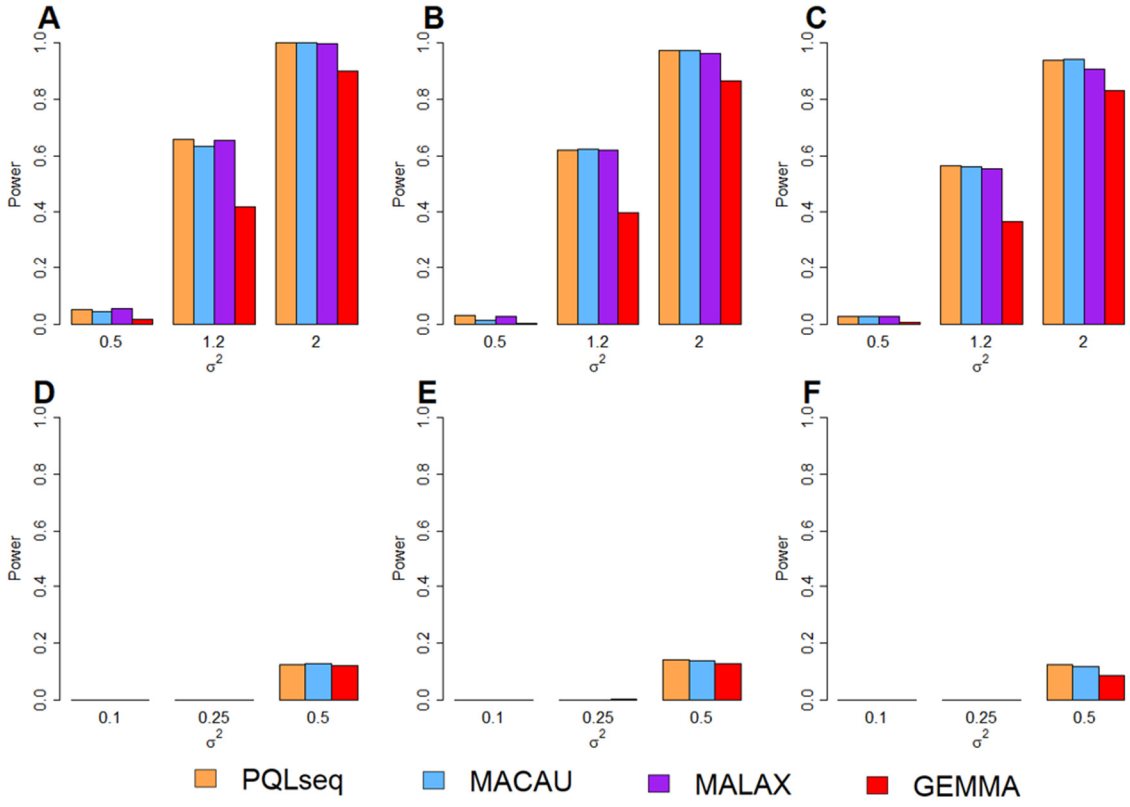
Supplementary Figure 10. PQLseq exhibits similar power as MACAU in BSseq and RNAseq based power simulations across a range of PVE and heritability values. The power results are obtained for PQLseq (orange), MACAU (blue), MALAX (purple) and GEMMA (red) based on 10% FDR in both BSseq based simulations (**A**, **B**, **C**) and RNAseq based simulations (**D**, **E**, **F**). Results are shown under different heritability values: h^2 = 0 (**A** and **D**), h^2 = 0.1 (**B** and **E**), or h^2=0.3 (**C** and **F**). The other parameter settings in the simulations are μ = 19, n = 100 and σ^2 = 1.2 for BSseq simulations; μ = 10, n = 100 and σ^2 = 0.25 for RNAseq simulations.



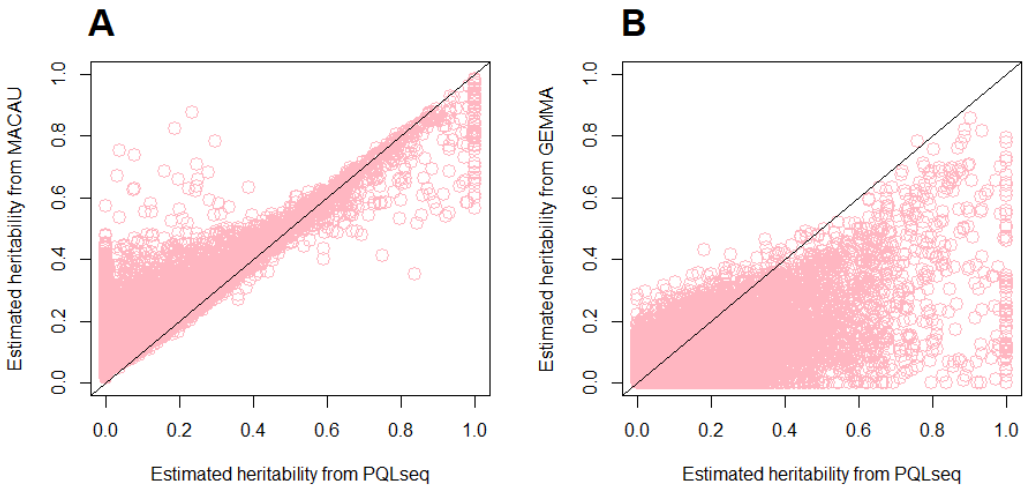
Supplementary Figure 11. PQLseq exhibits similar power as MACAU in BSseq and RNAseq based power simulations across a range of mean read counts μ and heritability values. The power results are obtained for PQLseq (orange), MACAU (blue), MALAX (purple) and GEMMA (red) based on 10% FDR in both BSseq based simulations (**A**, **B**, **C**) and RNAseq based simulations (**D**, **E**, **F**). Results are shown under different heritability values: h^2 = 0 (**A** and **D**), h^2 = 0.1 (**B** and **E**), or h^2=0.3 (**C** and **F**). The other parameter settings in the simulations are PVE = 0.25, n = 100 and σ^2 = 1.2 for BSseq simulations; PVE = 0.25, n = 100 and σ^2 = 0.25 for RNAseq simulations.



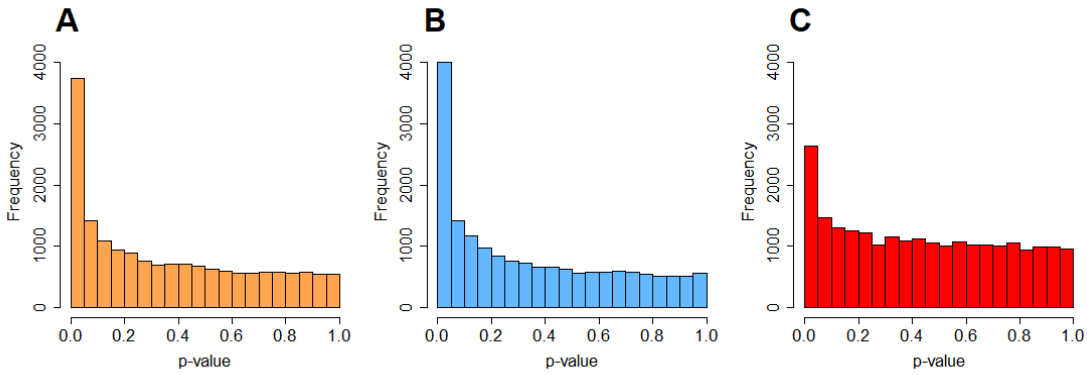
Supplementary Figure 12. PQLseq exhibits similar power as MACAU in BSseq and RNAseq based power simulations across a range of over-dispersion variance σ^2 and heritability values. The power results are obtained for PQLseq (orange), MACAU (blue), MALAX (purple) and GEMMA (red) based on 10% FDR in both BSseq based simulations (**A**, **B**, **C**) and RNAseq based simulations (**D**, **E**, **F**). Results are shown under different heritability values: h^2 = 0 (**A** and **D**), h^2 = 0.1 (**B** and **E**), or h^2=0.3 (**C** and **F**). The other parameter settings in the simulations are μ = 19, PVE = 0.25 and n = 100 for BSseq simulations; μ = 10, PVE = 0.25 and n = 100 for RNAseq simulations.



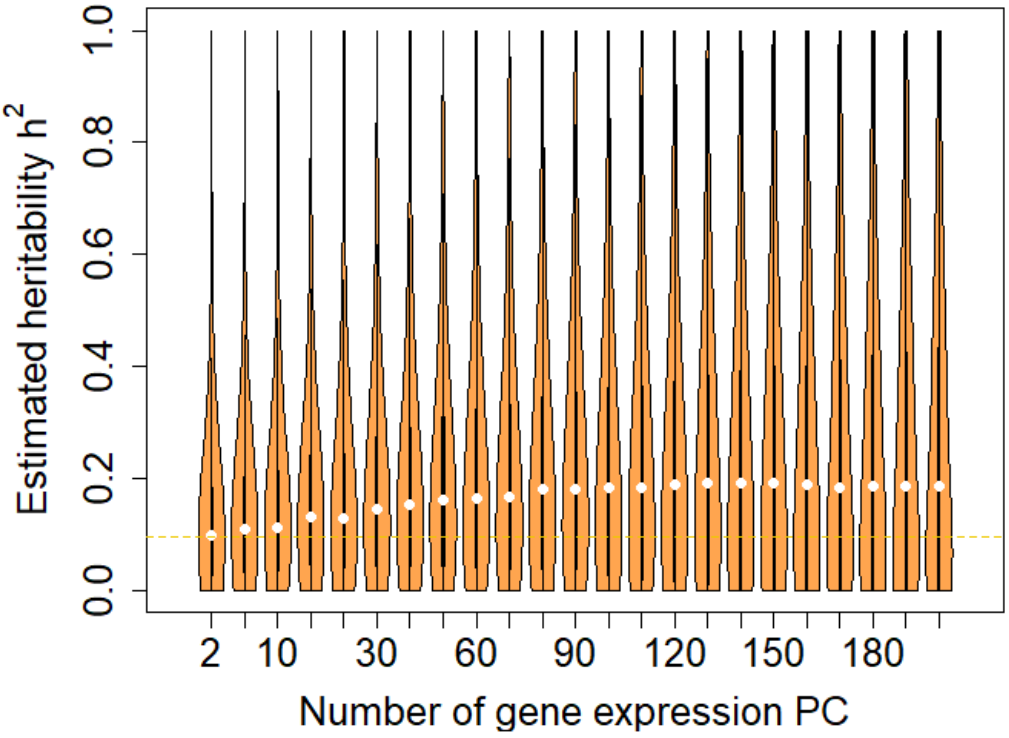
Supplementary Figure 13. Comparison of gene expression heritability estimates in the Hutterites RNAseq data. **A** shows that gene expression heritability estimates from MACAU are generally higher than that from PQLseq. **B** that gene expression heritability estimates from GEMMA are generally lower than that from PQLseq.



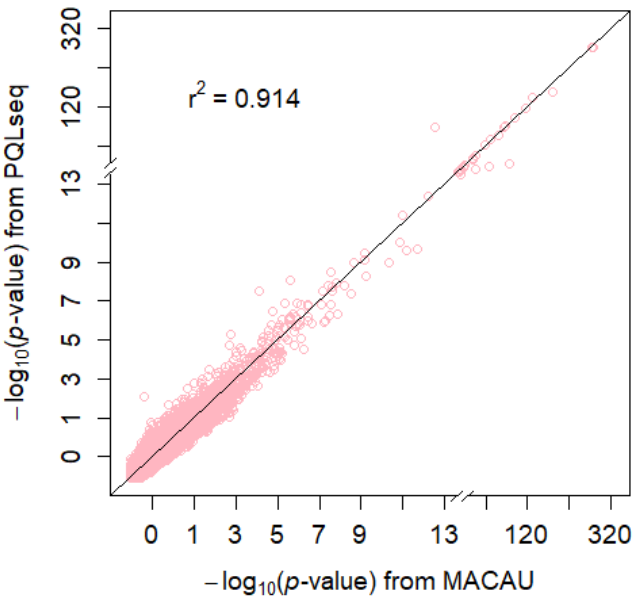
Supplementary Figure 14. Histogram of *p*-values from different methods for detecting differentially expressed genes in the Hutterites RNAseq data. The *p*-values are obtained from PQLseq (**A**), MACAU (**B**), and GEMMA (**C**) for detecting differentially expressed genes between genders.



Supplementary Figure 15. Heritability estimates from PQLseq in the Hutterites RNAseq data after adjusting for different numbers of gene expression principal components (PCs). Medium heritability estimates initially increase after adjusting for an increasing number of gene expression PCs, reach a peak, and gradually reduce afterwards.



Supplementary Figure 16. -log10(*p*-values) from PQLseq for detecting differentially expressed genes in different genders are highly correlated with that from MACAU in the Hutterites RNAseq data.



Simulations

We performed simulations to compare different methods. To make simulations as realistic as possible, we simulated either RNAseq data or BSseq data based on parameters inferred from two published data sets that include a RNAseq data set (Tung, et al., 2015) and a BSseq data set (Lea, et al., 2015). In the simulations, we varied the sample size (n) (n = 50, 100, 200, 300, or 500). To construct a relatedness matrix $\boldsymbol{K}$ in each of these sample simulations, we first obtained a real relatedness matrix from the published data (Lea, et al., 2015). We then constructed the relatedness matrix $\boldsymbol{K}$ by filling in its off-diagonal elements with randomly drawn off-diagonal elements from the real relatedness matrix following (Lea, et al., 2015). In cases where the resulting $\boldsymbol{K}$ was not positive definite, we used the *nearPD* function in R to find the closest positive definite matrix as the final $\boldsymbol{K}$. Besides n and $\boldsymbol{K}$, we also simulated a continuous predictor variable $\boldsymbol{x}$ from a standard normal distribution, and normalized the predictor $\boldsymbol{x}$ to have a zero mean and unit variance.

For RNAseq based simulations, in each simulation replicate, we simulated the total read count N_i for each individual from a discrete uniform distribution with a minimum (=1,770,083) and a maximum (=9,675,989) total read count (i.e., summation of read counts across all genes) equal to the minimum and maximum total read counts in the published RNAseq data (Tung, et al., 2015). We simulated 10,000 gene expression values and considered two general simulation settings. In the null settings, we simulated 10,000 non-differentially expressed (non-DE) genes to examine the gene expression heritability estimation accuracy and type I error control. In the alternative settings we simulated 1,000 DE genes and 9,000 non-DE genes to examine power. These non-DE or DE genes are simulated using the following procedure. Specifically, for each gene in turn, we simulated the genetic random effects $\boldsymbol{g}$ from a multivariate normal distribution with covariance $\boldsymbol{K}$. We simulated the environmental random effects $\boldsymbol{e}$ based on independent normal distributions. We then scaled the two sets of random effects to ensure a fixed value of heritability ($h^2 = \frac{V(g)}{V(g)+V(e)}$ = 0.0 or 0.1 or 0.3) and a fixed value of over-dispersion variance ($\sigma^2 = V(g) + V(e)$ = 0.1, 0.25 or 0.5) where the function $V(\bullet)$ denotes the sample variance. A heritability value of 0.1 and 0.3 correspond approximately to the median and upper 15% percentile of gene expression heritability estimates from the RNAseq data (Tung, et al., 2015). An over-dispersion variance value of 0.1, 0.25 and 0.5 correspond to approximately the lower quartile, median, and upper quartile of the over-dispersion variance inferred from the RNAseq data (Tung, et al., 2015). Afterwards, for non-DE genes, the genetic effects $\boldsymbol{g}$, environmental effects $\boldsymbol{e}$, and an intercept (μ) were then summed together to yield the latent variable $\log(\boldsymbol{\lambda}) = \mu + \boldsymbol{g} + \boldsymbol{e}$. Here, the intercept $\mu = \log(\frac{c}{\bar{N}})$ ensures an average gene count of c = 10, 50, or 100, where $\bar{N}$ is the average total read count across individuals. For DE genes, we used $\log(\boldsymbol{\lambda}) = \mu + \boldsymbol{x}\beta + \boldsymbol{g} + \boldsymbol{e}$ to yield the latent variable, where $\beta \sim N(0, \sigma_b^2)$ and σ_b^2 is set to ensure a fixed proportional of variance explained (PVE). That is, $\sigma_b^2 = \frac{PVE\ \sigma^2}{(1-PVE)V(\boldsymbol{x})}$, where PVE values were fixed to be 15%, 25%, or 35% to represent different effect sizes. Finally, we simulated the read counts based on a Poisson distribution with the Poisson rate being a product of the total read counts N_i and the latent variable λ_i; that is, $y_i \sim Poi(N_i\lambda_i)$ for the i'th individual. With the above procedure, we first simulated data under n = 100, h^2 = 0.1 and σ^2 = 0.25 (and PVE = 0.25 for DE

genes). We then varied one parameter at a time to generate different simulation scenarios. In each scenario, conditional on the sample size, total read counts etc., we performed 10 simulation replicates, each consisting of 10,000 genes.

For BSseq based simulations, in each simulation replicate, we simulated methylation values for 10,000 sites and considered two general simulation settings. In the null settings, we simulated 10,000 non-differentially methylated (non-DM) sites to examine the methylation level heritability estimation accuracy and type I error control. In the alternative settings we simulated 1,000 DM sites and 9,000 non-DM sites to examine power. These non-DM or DM sites are simulated using the following procedure. Specifically, for each site in turn, we simulated total read counts r_i for each individual i from a negative binomial distribution $r_i \sim NB(\mu, \theta)$ with μ = 18.80 and median θ = 2.49; the two parameter values correspond to the median estimates from the published BSseq data (Lea, et al., 2015). We then simulated the genetic random effects $\boldsymbol{g}$ and the environmental random effects $\boldsymbol{e}$ given a fixed heritability h^2 (0.1 or 0.3) and a fixed value of over-dispersion variance (σ^2 = 0.5, 1.2, or 2). Again, the over-dispersion variance values correspond to the lower quartile, median, and upper quartile of the over-dispersion variance inferred from the BSseq data (Lea, et al., 2015). For non-DM sites, the genetic effects $\boldsymbol{g}$, environmental effects $\boldsymbol{e}$, and an intercept (μ) were then summed together to yield the latent variable $\text{logit}(\boldsymbol{\pi}) = \mu + \boldsymbol{g} + \boldsymbol{e}$. Here, $\mu = \text{logit}(\frac{c}{\bar{r}})$ ensures an average number of methylated read counts being approximately c =5, 10 or 19, where $\bar{r}$ is the average total read count for the given site across individuals. For DM sites, we use $\text{logit}(\boldsymbol{\pi}) = \mu + \boldsymbol{x}\beta + \boldsymbol{g} + \boldsymbol{e}$ to yield the latent variable, where $\beta \sim N(0, \sigma_b^2)$ and σ_b^2 is set to ensure a fixed PVE. That is, $\sigma_b^2 = \frac{PVE\ \sigma^2}{(1-PVE)V(\boldsymbol{x})}$, where PVE values were set to be 15%, 25%, or 35% to represent different effect sizes. Finally, we simulated the methylated read counts based on a binomial distribution with a rate parameter determined by the total read counts r_i and the methylation proportion π_i; that is, $y_i \sim Bin(r_i, \pi_i)$ for the i'th individual. With the above procedure, we first simulated data under n = 100, $h^2 = 0.1$ and $\sigma^2 = 1.2$ (and PVE = 0.15 or PVE = 0.25 for DM sites). We then varied one parameter at a time to generate different scenarios. In each scenario, conditional on the sample size, total read counts etc., we performed 10 simulation replicates, each consisting of 10,000 sites.

We compared four different methods (PQLseq, MACAU, GEMMA, and MALAX) in the BSseq based simulations, and compared three different methods (PQLseq, MACAU and GEMMA) in the RNAseq based simulations as MALAX is only applicable for BSseq data. For GEMMA, we normalized data following previous recommendations (Lea, et al., 2015; Sun, et al., 2017). Specifically, for RNAseq data, for each gene in turn, we divided the number of read counts mapped to the gene by the total read depth, and quantile transformed the normalized data to a standard normal distribution. For BSseq data, we used "M" value transformation following (Du, et al., 2010) by dividing the number of methylated reads by the number of unmethylated reads followed by a log2-transformation. The normalized data is $log2\left(\frac{methylated\ reads+a}{unmethylated\ reads+a}\right)$, where a = 0.01 to avoid log transforming zero values.

Real Data Application

The published RNAseq data was collected from lymphoblastoid cell lines (LCLs) of 431 individuals from the Hutterites population in South Dakota, which is an isolated founder population (Cusanovich, et al., 2016). Libraries were created using the TruSeq Library Kit and samples were sequenced an Illumina HiSeq 2000 (50bp single end reads) in indexed pools of 12. Reads were trimmed for adaptors using Cutadapt (reads less than 5 bp discarded) then remapped to hg19 using bowtie indexed with gencode version 19 gene annotations (Langmead, et al., 2009; Martin, 2011). To remove mapping bias, autosomal reads were processed through WASP (van de Geijn, et al., 2015). Gene counts were quantified using HTSeq-count (Anders, et al., 2015) and verifyBamID was used to identify sample swaps (Jun, et al., 2012). Following these mapping and quality control steps, we obtained expression count measurements for 23,367 genes. We kept genes that have read counts greater than five in at least two individuals to focus on a final set of 17,312 genes. We also used the Hutterites pedigree information to compute the kinship coefficients between pairs of individuals and used them as the $\boldsymbol{K}$ matrix in the model. We then fitted a PMM for each gene in turn without any covariates to estimate gene expression heritability. For DE analysis, we used sex as the predictor variable (i.e. male vs female) to identify sex-associated genes. To compare performance among different methods for DE analysis, we permuted phenotype sex 20 times to obtain a null distribution. We used the null distribution to estimate the false discovery rate (FDR). Finally, to explore the influence of batch effects for heritability estimates in PQLseq, we extracted the top principal components (PCs) from the gene expression matrix and treated them as covariates in the model. We considered including a different number of top gene expression PCs that range from 2 to 200.

Inference for Generalized Mixed Models

Previously, we have developed a method, MACAU, that uses a Markov chain Monte Carlo (MCMC) sampling-based strategy to perform inference. In particular, we draw posterior samples from the GLMM and rely on the asymptotic normality of both the likelihood and the posterior distributions (Schwartz, 1965) to obtain the approximate maximum likelihood estimate $\hat{\beta}_j$ and its standard error se($\hat{\beta}_j$). With $\hat{\beta}_j$ and se($\hat{\beta}_j$), we construct approximate Wald test statistics and compute *p*-values for hypothesis testing. MACAU takes advantage of an auxiliary variable representation of the Poisson or binomial likelihood (Fruhwirth-Schnatter and Fruhwirth, 2010; Fruhwirth-Schnatter and Wagner, 2006; Scott, 2011) and recent linear algebra innovations for fitting linear mixed models (Lippert, et al., 2011; Zhou and Stephens, 2012; Zhou and Stephens, 2014). As a result, compared with a standard MCMC method, MACAU reduces the computational complexity of each MCMC iteration from cubic to quadratic with respect to the sample size, and is orders of magnitude faster than the popular Bayesian software MCMCglmm (Hadfield, 2010). However, despite the improvement, the early versions of MACAU are still computationally inefficient and are not readily applicable to large-scale genomic sequencing studies.

Our new method and software implementation, PQLseq, takes an alternative route for GLMM inference through the penalized quasi-likelihood (PQL) approach (Breslow and Clayton, 1993). Briefly, PQLseq employs an iterative numerical optimization procedure. In each iteration, we introduce a set of pseudo-data $\tilde{\boldsymbol{y}}$ to replace the originally observed data $\boldsymbol{y}$. The pseudo-data $\tilde{\boldsymbol{y}}$ is obtained based on a second order Taylor expansion using the conditional distribution $P(y_i|\boldsymbol{g},\boldsymbol{e})$ using the first and second order moments $\mathrm{E}(y_i|\boldsymbol{g},\boldsymbol{e})$ and $V(y_i|\boldsymbol{g},\boldsymbol{e})$, both evaluated at the current estimates of the fixed coefficients as well as the random effects $\boldsymbol{g}$ and $\boldsymbol{e}$. With the pseudo-data, the complex GLMM likelihood function for the original data $\boldsymbol{y}$ is replaced by a much simpler LMM likelihood function for the pseudo-data $\tilde{\boldsymbol{y}}$, thereby alleviating much of the computational burden associated with GLMM. With pseudo-data $\tilde{\boldsymbol{y}}$, we can perform inference and update parameters using the standard average information (AI) algorithm for LMMs (Chen, et al., 2016; Gilmour, et al., 1995; Yang, et al., 2011). By iterating between the approximation step of obtaining the pseudo-data $\tilde{\boldsymbol{y}}$ and the inference step of updating the parameter estimates via the AI algorithm, PQLseq allows us to perform inference in a computationally efficient fashion.

Below, we describe the estimation and inference procedure in detail. To facilitate description, we introduce $\mathbf{W}$ as the n by c matrix of covariates, and $\mathbf{x}$ as the n-vector of the predictor variable of interest.

First, observations y_i $(i = 1, 2, \cdots, n)$ are independent conditional on the unobserved random effects $\boldsymbol{u} = \boldsymbol{g} + \boldsymbol{e}$ and the fixed effects $\mathbf{W}\boldsymbol{\alpha} + \mathbf{x}\beta$, with conditional mean $\mathrm{E}(y_i|\boldsymbol{u},\boldsymbol{\alpha},\beta) = \mu_i = g^{-1}(\boldsymbol{w}_i^T\boldsymbol{\alpha} + x_i\beta + u_i)$ and conditional variance $\mathrm{V}(y_i|\boldsymbol{u},\boldsymbol{\alpha},\beta) = v(\mu_i)$, where $g(\cdot)$ is the usual link function (i.e. log link for PMM and logit link for BMM) and $v(\cdot)$ is the variance function (i.e. $v(t) = t$ for the PMM and BMM). We use these two conditional moments to obtain the quasi-likelihood for i-th individual, $ql_i(\boldsymbol{\alpha},\beta|\tilde{\boldsymbol{u}}) = \int_{y_i}^{\mu_i} \frac{y_i - t}{v(t)} dt$, which serves as an approximation for the conditional likelihood. The joint likelihood function can thus be approximated by the joint quasi-likelihood function

$$ql(\boldsymbol{\alpha}, \beta, \sigma^2, h^2) = \log \int \left(\prod_{i=1}^{n} ql_i(\boldsymbol{u}, \boldsymbol{\alpha}, \beta) \right) P(\boldsymbol{u}|\sigma^2, h^2) d\boldsymbol{u}.$$

We use Laplace approximation to further approximate the above function and obtain

$$\widetilde{ql}(\boldsymbol{\alpha}, \beta, \sigma^2, h^2) = \frac{1}{2}\log|\mathbf{VD} + \mathbf{I}| + \sum_{i=1}^{n} ql_i(\widetilde{\boldsymbol{u}}, \boldsymbol{\alpha}, \beta) - \frac{1}{2}\widetilde{\boldsymbol{u}}^T\mathbf{V}^{-1}\widetilde{\boldsymbol{u}}, \qquad (1)$$

where $\mathbf{V} = \sigma^2 h^2 \mathbf{K} + \sigma^2(1-h^2)\mathbf{I}$, $\widetilde{\boldsymbol{u}} = \underset{\boldsymbol{u}}{\operatorname{argmax}} \left(\sum_{i=1}^{n} ql_i(\boldsymbol{\alpha}, \beta|\boldsymbol{u}) - \frac{1}{2}\boldsymbol{u}^T\mathbf{V}^{-1}\boldsymbol{u} \right)$ and $\mathbf{D} = \operatorname{diag}(1/g'(\mu_i))$ is a diagonal weight matrix.

We treat the approximated quasi-likelihood function $\widetilde{ql}$ in equation (1) as the target function. And we obtain estimates for $(\boldsymbol{u}, \boldsymbol{\alpha}, \beta)$ and (σ^2, h^2) alternately from equation (1).

Specifically, we first obtain estimates for $(\boldsymbol{u}, \boldsymbol{\alpha}, \beta)$ conditional on the current estimates of (σ^2, h^2). To do so, following (Breslow and Clayton, 1993; Gilmour, et al., 1995), we assume that the iterative weights vary slowly with respect to the conditional mean; that is

$$\frac{\partial \mathbf{D}}{\partial \mu_i} \approx 0.$$

We then obtain the first order derivatives with respect to either $(\boldsymbol{\alpha}, \beta)$ or $\boldsymbol{u}$, and set the two first order derivatives to zero; that is

$$(\mathbf{W}, \boldsymbol{x})^T \mathbf{D}\Delta(\boldsymbol{y} - \boldsymbol{\mu}) = \mathbf{0}, \qquad (2)$$

$$\mathbf{D}\Delta(\boldsymbol{y} - \boldsymbol{\mu}) - \mathbf{V}^{-1}\boldsymbol{u} = \mathbf{0}, \qquad (3)$$

Where $\boldsymbol{\mu} = (\mu_1, \cdots, \mu_n)$ and $\Delta = \operatorname{diag}\{g'(\mu_i)\}$.

We now define the pseudo-data

$$\widetilde{\mathbf{Y}} = \eta_i + g'(\mu_i)(y_i - \mu_i), \qquad (4)$$

and our equation (2) becomes

$$\mathbf{u} = \mathbf{VH}^{-1}\left[\widetilde{\mathbf{Y}} - (\mathbf{W}, \boldsymbol{x})\begin{pmatrix}\boldsymbol{\alpha}\\ \beta\end{pmatrix}\right], \qquad (5)$$

where $\mathbf{H} = \mathbf{D}^{-1} + \mathbf{V}$. Substituting equation (5) into equation (3), we can obtain the estimates

$$\begin{pmatrix}\widehat{\boldsymbol{\alpha}}\\ \hat{\beta}\end{pmatrix} = [(\mathbf{W}, \boldsymbol{x})^T\mathbf{H}^{-1}(\mathbf{W}, \boldsymbol{x})]^{-1}(\mathbf{W}, \boldsymbol{x})^T\boldsymbol{H}^{-1}\widetilde{\mathbf{Y}}, \qquad (6)$$

and

$$\widehat{\mathbf{u}} = \mathbf{VH}^{-1}\left[\widetilde{\mathbf{Y}} - (\mathbf{W}, \boldsymbol{x})\begin{pmatrix}\widehat{\boldsymbol{\alpha}}\\ \hat{\beta}\end{pmatrix}\right], \qquad (7)$$

Both are conditional on the variance component estimates (σ^2, h^2)

Next, we obtain estimates for the variance components (σ^2, h^2) conditional on the current estimates of $(\boldsymbol{u}, \boldsymbol{\alpha}, \beta)$. To do so, we first define the transformed variance components $\tau_1 =$

$\sigma^2 h^2$ and $\tau_2 = \sigma^2(1-h^2)$. We then integrate out the fix effects $\boldsymbol{\alpha}$ and β in equation (1) to obtain the restricted likelihood function as

$$\widetilde{ql}_R(\tau_1, \tau_2) = c_R - \frac{1}{2}\log|\mathbf{H}| - \frac{1}{2}\log|(\mathbf{W}, \boldsymbol{x})^T\mathbf{H}^{-1}(\mathbf{W}, \boldsymbol{x})| - \frac{1}{2}\widetilde{\mathbf{Y}}^T\mathbf{P}\widetilde{\mathbf{Y}},$$

where $\mathbf{P} = \mathbf{H}^{-1} - \mathbf{H}^{-1}(\mathbf{W}, \boldsymbol{x})^T\left((\mathbf{W}, \boldsymbol{x})^T\mathbf{H}^{-1}(\mathbf{W}, \boldsymbol{x})\right)^{-1}(\mathbf{W}, \boldsymbol{x})\mathbf{H}^{-1}$, and c_R is a constant. We use the AI algorithm to obtain variance component estimates. In particular, we obtain the first derivatives as

$$\frac{\partial \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_1} = \frac{1}{2}\left\{\widetilde{\mathbf{Y}}^T\mathbf{PKP}\widetilde{\mathbf{Y}} - tr(\mathbf{PK})\right\}, \quad \frac{\partial \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_2} = \frac{1}{2}\left\{\widetilde{\mathbf{Y}}^T\mathbf{PIP}\widetilde{\mathbf{Y}} - tr(\mathbf{PI})\right\},$$

and the second derivatives as

$$\frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_1 \partial \tau_1} = \frac{1}{2}tr(\mathbf{KPKP}) - \widetilde{\mathbf{Y}}^T\mathbf{PKPKP}\widetilde{\mathbf{Y}}, \quad \frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_1 \partial \tau_2} = \frac{1}{2}tr(\mathbf{KPIP}) - \widetilde{\mathbf{Y}}^T\mathbf{PKPIP}\widetilde{\mathbf{Y}},$$

$$\frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_2 \partial \tau_2} = \frac{1}{2}tr(\mathbf{IPIP}) - \widetilde{\mathbf{Y}}^T\mathbf{PIPIP}\widetilde{\mathbf{Y}}, \qquad \frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_2 \partial \tau_1} = \frac{1}{2}tr(\mathbf{IPKP}) - \widetilde{\mathbf{Y}}^T\mathbf{PIPKP}\widetilde{\mathbf{Y}}.$$

The second derivatives constitute the observed information matrix. Because the elements in the expected information matrix are

$$E\left[\frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_1 \partial \tau_1}\right] = -\frac{1}{2}tr(\mathbf{KPKP}), \quad E\left[\frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_1 \partial \tau_2}\right] = -\frac{1}{2}tr(\mathbf{KPIP}),$$

$$E\left[\frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_2 \partial \tau_2}\right] = -\frac{1}{2}tr(\mathbf{IPIP}), \qquad E\left[\frac{\partial^2 \widetilde{ql}_R(\sigma^2,h^2)}{\partial \tau_2 \partial \tau_1}\right] = -\frac{1}{2}tr(\mathbf{IPKP}),$$

we can obtain the average information (AI) matrix as an average of the above two matrices; that is

$$\mathbf{AI} = \begin{bmatrix} \widetilde{\mathbf{Y}}^T\mathbf{PKPKP}\widetilde{\mathbf{Y}} & \widetilde{\mathbf{Y}}^T\mathbf{PKPIP}\widetilde{\mathbf{Y}} \\ \widetilde{\mathbf{Y}}^T\mathbf{PIPKP}\widetilde{\mathbf{Y}} & \widetilde{\mathbf{Y}}^T\mathbf{PIPIP}\widetilde{\mathbf{Y}} \end{bmatrix}.$$

With the first and second order derivatives, we can perform Newton-Raphson update with the AI algorithm and obtain estimates for (τ_1, τ_2), which in turn leads to estimates of (σ^2, h^2).

As a summary, PQLseq implements the PQL algorithm that consists of the following steps:

1. Initialize the parameters, $\boldsymbol{\alpha}^{(0)}, \beta^{(0)}, \boldsymbol{\tau}^{(0)} = \left(\tau_1^{(0)}, \tau_2^{(0)}\right)^T$, and obtain the pseudo-data $\widetilde{\mathbf{Y}}^{(0)}$ as in equation (4). Set $t = 1$.
2. Update $\boldsymbol{\tau}^{(t)} = \boldsymbol{\tau}^{(t-1)} + \mathbf{AI}^{-1}\left(\frac{\partial ql_R(\sigma^2,h^2)}{\partial \boldsymbol{\tau}}\right)$;
3. Update $\boldsymbol{\alpha}^{(t)}, \beta^{(t)}$ and $\boldsymbol{u}^{(t)}$ with $\boldsymbol{\tau}^{(t)}$ and $\widetilde{\mathbf{Y}}^{(t-1)}$ as in equations (6) and (7);
4. Update $\widetilde{\mathbf{Y}}^{(t)}$ using the $\boldsymbol{\alpha}^{(t)}, \beta^{(t)}$ and $\boldsymbol{u}^{(t)}$ as in equation (4);
5. Set $t = t + 1$, and repeat steps 2-4 until convergence.

Once we obtain parameter estimates $(\boldsymbol{\alpha}, \beta, \sigma^2, h^2)$, we can construct the Wald test based on

$$\hat{\beta} = (\boldsymbol{x}^T\mathbf{P}_c\boldsymbol{x})^{-1}\boldsymbol{x}^T\mathbf{P}_c\widetilde{\mathbf{Y}} \text{ and } \mathrm{Var}(\hat{\beta}) = (\boldsymbol{x}^T\mathbf{P}_c\boldsymbol{x})^{-1},$$

where $\mathbf{P} = \mathbf{H}^{-1} - \mathbf{H}^{-1}\mathbf{W}^{T}(\mathbf{W}^{T}\mathbf{H}^{-1}\mathbf{W}^{T})^{-1}\mathbf{W}^{T}\mathbf{H}^{-1}$.

REFERENCE

Anders, S., Pyl, P.T. and Huber, W. (2015) HTSeq-a Python framework to work with high-throughput sequencing data, *Bioinformatics*, **31**, 166-169.

Breslow, N.E. and Clayton, D.G. (1993) Approximate Inference In Generalized Linear Mixed Models, *J Am Stat Assoc*, **88**, 9-25.

Chen, H.*, et al.* (2016) Control for Population Structure and Relatedness for Binary Traits in Genetic Association Studies via Logistic Mixed Models, *The American Journal of Human Genetics*, **98**, 653--666.

Cusanovich, D.A.*, et al.* (2016) Integrated analyses of gene expression and genetic association studies in a founder population, *Hum Mol Genet*, **25**, 2104-2112.

Du, P.*, et al.* (2010) Comparison of Beta-value and M-value methods for quantifying methylation levels by microarray analysis, *Bmc Bioinformatics*, **11**, 587.

Fruhwirth-Schnatter, S. and Fruhwirth, R. (2010) *Data Augmentation and MCMC for Binary and Multinomial Logit Models*. Statistical Modelling And Regression Structures:Festschrift in Honour of Ludwig Fahrmeir. Springer, New York.

Fruhwirth-Schnatter, S. and Wagner, H. (2006) Auxiliary mixture sampling for parameter-driven models of time series of counts with applications to state space modelling, *Biometrika*, **93**, 827-841.

Gilmour, A.R., Thompson, R. and Cullis, B.R. (1995) Average information REML: An efficient algorithm for variance parameter estimation in linear mixed models, *Biometrics*, **51**, 1440-1450.

Hadfield, J.D. (2010) MCMC Methods for Multi-Response Generalized Linear Mixed Models: The MCMCglmm R Package, *J Stat Softw*, **33**, 1-22.

Jun, G.*, et al.* (2012) Detecting and Estimating Contamination of Human DNA Samples in Sequencing and Array-Based Genotype Data, *Am J Hum Genet*, **91**, 839-848.

Langmead, B.*, et al.* (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome, *Genome Biol*, **10**, R25.

Lea, A.J.*, et al.* (2015) A flexible, efficient binomial mixed model for identifying differential DNA methylation in bisulfite sequencing data, *Plos Genet*, **11**, e1005650.

Lippert, C.*, et al.* (2011) FaST linear mixed models for genome-wide association studies, *Nat Methods*, **8**, 833-835.

Martin, M. (2011) Cutadapt removes adapter sequences from high-throughput sequencing reads, *Marcel Martin*, **17**, 10-12.

Schwartz, L. (1965) On Bayes procedures, *Zeitschrift f{\"u}r Wahrscheinlichkeitstheorie und Verwandte Gebiete*, **4**, 10--26.

Scott, S.L. (2011) Data augmentation, frequentist estimation, and the Bayesian analysis of multinomial logit models, *Stat Pap*, **52**, 87-109.

Sun, S.Q.*, et al.* (2017) Differential expression analysis for RNAseq using Poisson mixed models, *Nucleic Acids Res*, **45**, e106.

Tung, J.*, et al.* (2015) The genetic architecture of gene expression levels in wild baboons, *Elife*, **4**, e04729.

van de Geijn, B.*, et al.* (2015) WASP: allele-specific software for robust molecular quantitative trait locus discovery, *Nat Methods*, **12**, 1061-1063.

Yang, J.A.*, et al.* (2011) GCTA: A Tool for Genome-wide Complex Trait Analysis, *Am J Hum Genet*, **88**, 76-82.

Zhou, X. and Stephens, M. (2012) Genome-wide efficient mixed-model analysis for association studies, *Nat Genet*, **44**, 821-824.

Zhou, X. and Stephens, M. (2014) Efficient multivariate linear mixed model algorithms for genome-wide association studies, *Nat Methods*, **11**, 407-409.