

BOLT-LMM v2.0 User Manual

Po-Ru Loh

March 13, 2015

Contents

1	Overview	2
1.1	BOLT-LMM mixed model association testing	2
1.2	BOLT-REML variance components analysis	3
2	Installation	3
2.1	Running BOLT-LMM and BOLT-REML	4
2.2	Examples	4
2.3	Help	4
3	Computing requirements	5
3.1	Operating system	5
3.2	Memory	5
3.3	Running time	5
3.3.1	Multi-threading	5
4	Input/output file naming conventions	5
4.1	Automatic gzip [de]compression	5
4.2	Arrays of input files and covariates	6
5	Input	6
5.1	Genotypes	6
5.1.1	Reference genetic maps	6
5.1.2	Imputed SNP dosages	6
5.2	Phenotypes	8
5.3	Covariates	8
5.4	Missing data treatment	9
5.5	Genotype QC	9
5.6	User-specified filters	9

6	Association analysis (BOLT-LMM)	9
6.1	Mixed model association tests	9
6.2	BOLT-LMM mixed model association options	10
6.2.1	Restricting SNPs used in the mixed model	10
6.3	Standard linear regression	10
7	Variance components analysis (BOLT-REML)	10
7.1	Multiple variance components	11
7.2	Multiple traits	11
7.3	Initial variance parameter guesses	11
7.4	Trading a little accuracy for speed	11
8	Output	12
8.1	BOLT-LMM association test statistics	12
8.2	BOLT-REML output and logging	12
9	Change log	13
10	Website and contact info	13
11	License	13

1 Overview

The BOLT-LMM software package currently consists of two main algorithms, the BOLT-LMM algorithm for mixed model association testing, and the BOLT-REML algorithm for variance components analysis (i.e., partitioning of SNP-heritability and estimation of genetic correlations).

1.1 BOLT-LMM mixed model association testing

The BOLT-LMM algorithm computes statistics for testing association between phenotype and genotypes using a linear mixed model (LMM) [1]. By default, BOLT-LMM assumes a Bayesian mixture-of-normals prior for the random effect attributed to SNPs other than the one being tested. This model generalizes the standard “infinitesimal” mixed model used by existing mixed model association methods (e.g., EMMAX [2], FaST-LMM [3–6], GEMMA [7], GRAMMAR-Gamma [8], GCTA-LOCO [9]), providing an opportunity for increased power to detect associations while controlling false positives. Additionally, BOLT-LMM applies algorithmic advances to compute mixed model association statistics much faster than existing methods, both when using the Bayesian mixture model and when specialized to standard mixed model association. BOLT-LMM is described in ref. [1]:

Loh P-R, Tucker G, Bulik-Sullivan BK, Vilhjálmsdóttir BJ, Finucane HK, Salem RM, Chasman DI, Ridker PM, Neale BM, Berger B, Patterson N, and Price AL. Efficient

Bayesian mixed model analysis increases association power in large cohorts. *Nature Genetics*, 2015.

1.2 BOLT-REML variance components analysis

The BOLT-REML algorithm estimates heritability explained by genotyped SNPs and genetic correlations among multiple traits measured on the same set of individuals. Like the GCTA software [10], BOLT-REML applies variance components analysis to perform these tasks, supporting both multi-component modeling to partition SNP-heritability and multi-trait modeling to estimate correlations. BOLT-REML applies a Monte Carlo algorithm that is much faster than standard methods for variance components analysis (e.g., GCTA) at large sample sizes. BOLT-REML is described in ref. [11]:

Loh P-R, Bhatia G, Gusev A, Finucane HK, Bulik-Sullivan BK, Pollack SJ, PGC-SCZ Working Group, de Candia TR, Lee SH, Wray NR, Kendler KS, O'Donovan MC, Neale BM, Patterson N, and Price AL. Contrasting regional architectures of schizophrenia and other complex diseases using fast variance components analysis. *bioRxiv*, 2015.

2 Installation

We provide a standalone (i.e., statically linked) 64-bit Linux executable, `bolt`, which we have tested on several Linux systems. We strongly recommend using this static executable because it is well-optimized and no further installation is required.

If you wish to compile your own version of the BOLT-LMM software from the source code (in the `src/` subdirectory, provided in a separate download licensed under GNU GPLv3), you will need to ensure that library dependencies are fulfilled and will need to make appropriate modifications to the `Makefile`:

- Library dependencies:
 - BLAS/LAPACK numerical libraries. The speed of the BOLT-LMM software depends critically on the efficiency of the BLAS/LAPACK implementation it is linked against. We recommend the Intel Math Kernel Library (MKL) if available; otherwise, ATLAS may be a good alternative.
 - Boost C++ libraries. BOLT-LMM links against the Boost `program_options` and `iostreams` libraries, which need to be installed after downloading and unzipping Boost.
 - NLOpt numerical optimization library [12].
- `Makefile`: Paths to libraries need to be modified appropriately. Note that the released version of the `Makefile` does not set the flag `-DUSE_MKL_MALLOC`. This flag turns on the Intel MKL's fast memory manager (replacing calls to `_mm_malloc` with `mk1_malloc`),

which may improve memory performance, but we have observed crashes on some systems when using `mkl_malloc`.

For reference, the provided `bolt` executable was created on the Harvard Medical School “Orchestra” research computing cluster using Intel Composer XE 12.0.5 (MKL 10.3.5) and the Boost 1.42 and NLOpt 2.4.2 libraries by invoking `make cluster=orchestra linking=static`.

2.1 Running BOLT-LMM and BOLT-REML

To run the `bolt` executable, simply invoke `./bolt` on the Linux command line (within the BOLT-LMM install directory) with parameters in the format `--optionName=optionValue`.

2.2 Examples

The `example/` subdirectory contains a bash script `run_example.sh` that demonstrates basic use of BOLT-LMM on a small example data set. Likewise, `run_example_reml2.sh` demonstrates BOLT-REML.

- A minimal BOLT-LMM invocation looks like:

```
./bolt --bfile=geno --phenoFile=pheno.txt --phenoCol=phenoName  
      --lmm --LDscoresFile=tables/LDSCORE.1000G_EUR.tab.gz  
      --statsFile=stats.tab
```

- A minimal BOLT-REML invocation looks like:

```
./bolt --bfile=geno --phenoFile=pheno.txt --phenoCol=phenoName  
      --reml --modelSnps=modelSnps.txt
```

To perform multi-trait BOLT-REML (i.e., estimate genetic correlations), provide multiple `--phenoCol=phenoN` arguments.

2.3 Help

To get a list of basic options, run:

```
./bolt -h
```

To get a complete list of basic and advanced options, run:

```
./bolt --helpFull
```

3 Computing requirements

3.1 Operating system

At the current time we have only compiled and tested BOLT-LMM on Linux computing environments; however, the source code is available if you wish to try compiling BOLT-LMM for a different operating system.

3.2 Memory

For typical data sets ($M, N > 10,000$), BOLT-LMM and BOLT-REML use approximately $MN/4$ bytes of memory, where M is the number of SNPs and N is the number of individuals. More precisely:

- M = # of SNPs in `bim` file(s) that satisfy all of the conditions:
 - not listed in any `--exclude` file
 - passed QC filter for missingness
 - listed in `--modelSnps` file(s), if specified
- N = # of individuals in `fam` file and not listed in any `--remove` file (but pre-QC; i.e., N includes individuals filtered due to missing genotypes or covariates)

3.3 Running time

In practice, BOLT-LMM and BOLT-REML have running times that scale roughly with $MN^{1.5}$. Our largest analyses of real data ($M = 600K$ SNPs, $N = 60K$ individuals) took ≈ 1 day using a single computational core. We have also tested BOLT-LMM on simulated data sets containing up to $N = 480K$ individuals; for more details, please see the BOLT-LMM manuscript [1].

3.3.1 Multi-threading

On multi-core machines, running time can be reduced by invoking multi-threading using the `--numThreads` option.

4 Input/output file naming conventions

4.1 Automatic gzip [de]compression

The BOLT-LMM software assumes that input files ending in `.gz` are gzip-compressed and automatically decompresses them on-the-fly (i.e., without creating a temporary file). Similarly, BOLT-LMM writes gzip-compressed output to any output file ending in `.gz`.

4.2 Arrays of input files and covariates

Arrays of sequentially-numbered input files and covariates can be specified by the shorthand `{i:j}`. For example,

```
data.chr{1:22}.bim
```

is interpreted as the list of files

```
data.chr1.bim, data.chr2.bim, ..., data.chr22.bim
```

5 Input

5.1 Genotypes

The BOLT-LMM software takes genotype input in PLINK [13] binary format (`bed/bim/fam`). For file conversion and data manipulation in general, we highly recommend the PLINK2 software [14], which is providing a comprehensive, much more efficient update to PLINK.

If all genotypes are contained in a single `bed/bim/fam` file triple with the same file prefix, you may simply use the command line option `--bfile=prefix`. Genotypes may also be split into multiple `bed` and `bim` files containing consecutive sets of SNPs (e.g., one `bed/bim` file pair per chromosome) either by using multiple `--bed` and `--bim` invocations or by using the file array shorthand described above (e.g., `--bim=data.chr{1:22}.bim`).

5.1.1 Reference genetic maps

The BOLT-LMM package includes reference maps that you can use to interpolate genetic map coordinates from SNP physical (base pair) positions in the event that your PLINK `bim` file does not contain genetic coordinates (in units of morgans). (The BOLT-LMM association testing algorithm uses genetic positions to prevent proximal contamination; BOLT-REML does not use this information.) To use a reference map, use the option

```
--geneticMapFile=tables/genetic_map_hg##.txt.gz
```

selecting the build (hg17, hg18, or hg19) corresponding to the physical coordinates of your `bim` file. You may use the `--geneticMapFile` option even if your PLINK `bim` file does contain genetic coordinates; in this case, the genetic coordinates in the `bim` file will be ignored, and interpolated coordinates will be used instead.

5.1.2 Imputed SNP dosages

As of version 1.1, the BOLT-LMM association testing algorithm supports computation of mixed model association statistics at an arbitrary number of imputed SNPs (with real-valued “dosages” rather than hard-called genotypes) using a mixed model built on a subset of hard-called genotypes. (BOLT-REML variance components analysis does not support dosage input.) This approach requires only a trivial amount of additional computation and no additional RAM, as it simply applies

a genome scan (as in GRAMMAR-Gamma [8]) of real-valued dosage SNPs against the residual phenotypes that BOLT-LMM already computes. We expect that using a mixed model built on only a subset of $\approx 500\text{K}$ hard-called genotypes should sacrifice almost no power while retaining the computational efficiency of BOLT-LMM.

If you have only imputed SNP data on hand, you will need to pre-process your data set to create a subset of hard-called SNPs in PLINK format for BOLT-LMM. We suggest the following procedure.

1. Determine a high-confidence set of SNPs (e.g., based on IMPUTE2 INFO score) at which to create an initial hard-call set.
2. Create hard-called genotypes at these SNPs in PLINK format.
3. Use PLINK to LD prune to $\approx 500\text{K}$ SNPs (via `--indep-pairwise 50 5 r2thresh` for an appropriate *r2thresh*).
4. Run BOLT-LMM using the final hard-called SNPs as the `--bfile` (or `bed/bim/fam`) argument, specifying the imputed SNPs as additional association test SNPs using one of the formats below.

Imputed SNPs in dosage format. This input format consists of one or more `--dosageFile` parameters specifying files that contain real-valued genotype expectations at imputed SNPs. Each line of a `--dosageFile` should be formatted as follows:

```
rsID    chr    pos    allele1    allele0    [dosage = E[#allele1]] x N
```

Missing (i.e., uncalled) dosages can be specified with `-9`. You will also need to provide one additional `--dosageFidIidFile` specifying the PLINK FIDs and IIDs of samples that the dosages correspond to. See the `example/` subdirectory for an example.

Imputed SNPs in IMPUTE2 format. You may also specify imputed SNPs as output by the IMPUTE2 software [15]. The IMPUTE2 genotype file format is as follows:

```
snpID    rsID    pos    allele1    allele0    [p(11) p(10) p(00)] x N
```

(BOLT-LMM ignores the `snpID` field.) Here, instead of dosages, each genotype entry contains individual probabilities of the individual being homozygous for `allele1`, heterozygous, and homozygous for `allele0`. The three probabilities need not sum to 1, allowing for genotype uncertainty; if the sum of the probabilities is less than the `--impute2CallThresh` parameter, BOLT-LMM treats the genotype as missing.

To compute association statistics at a list of files containing IMPUTE2 SNPs, you may list the files within a `--impute2FileList` file. Each line of this file should contain two entries: a chromosome number followed by an IMPUTE2 genotype file containing SNPs from that chromosome. You will also need to provide one additional `--impute2FidIidFile` specifying the PLINK FIDs and IIDs of samples that the IMPUTE2 genotypes correspond to. See the `example/` subdirectory for an example.

Imputed SNPs in 2-dosage format. You may also specify imputed SNPs as output by the Riecopili pipeline and `plink2 --dosage format=2`. This file format consists of file pairs: (1) PLINK map files containing information about SNP locations; and (2) genotype probability files in the 2-dosage format, which consists of a header line

```
SNP    A1    A2    [FID IID] x N
```

followed by one line per SNP in the format

```
rsID    allele1    allele0    [p(11) p(10)] x N
```

The third genotype probability for each entry is assumed to be $p(00) = 1 - p(11) - p(10)$ (unlike with the IMPUTE2 format).

To compute association statistics at SNPs in a list of 2-dosage files, you may list the files within a `--dosage2FileList` file. Each line of this file should contain two entries: a PLINK map file followed by the corresponding genotype file containing probabilities for those SNPs. (As usual, if either file ends with `.gz`, it is automatically unzipped; otherwise it is assumed to be plain text.) See the `example/` subdirectory for an example.

5.2 Phenotypes

Phenotypes may be specified in either of two ways:

- `--phenoUseFam`: This option tells BOLT-LMM and BOLT-REML to use the last (6th) column of the `fam` file as the phenotypes. This column must be numeric, so case-control phenotypes should be 0, 1 coded and missing values should be indicated with `-9`.
- `--phenoFile` and `--phenoCol`: Alternatively, phenotypes may be provided in a separate whitespace-delimited file (specified with `--phenoFile`) with the first line containing column headers and subsequent lines containing records, one per individual. The first two columns must be `FID` and `IID` (the PLINK identifiers of an individual). Any number of columns may follow; the column containing the phenotype to analyze is specified with `--phenoCol`. Values of `-9` and `-NA` are interpreted as missing data. All other values in the column should be numeric. The records in lines following the header line need not be in sorted order and need not match the individuals in the genotype data (i.e., `fam` file); BOLT-LMM and BOLT-REML will analyze only the individuals in the intersection of the genotype and phenotype files and will output a warning if these sets do not match.

5.3 Covariates

Covariate data may be specified in a file (`--covarFile`) with the same format as the alternate phenotype file described above. (The same file may be used for both phenotypes and covariates.) Each covariate to be used must be specified using either a `--covarCol` (for categorical covariates) or a `--qCovarCol` (for quantitative covariates) option. Categorical covariate values are allowed to be any text strings not containing whitespace; each unique text string in a column

corresponds to a category. Quantitative covariate values must be numeric (with the exception of NA). In either case, values of `-9` and `-NA` are interpreted as missing data. If groups of covariates of the same type are numbered sequentially, they may be specified using array shorthand (e.g., `--qCovarCol=PC{1:10}` for columns `PC1`, `PC2`, ..., `PC10`).

5.4 Missing data treatment

Individuals with missing phenotypes are ignored. By default, individuals with any missing covariates are also ignored; this approach is commonly used and referred to as “complete case analysis.” As an alternative, we have also implemented the “missing indicator method” (via the `--covarUseMissingIndic` option), which adds indicator variables demarcating missing status as additional covariates. Missing genotypes are replaced with per-SNP averages.

5.5 Genotype QC

BOLT-LMM and BOLT-REML automatically filter SNPs and individuals with missing rates exceeding thresholds of 0.1. These thresholds may be modified using `--maxMissingPerSnp` and `--maxMissingPerIndiv`. Note that filtering is **not** performed based on minor allele frequency or deviation from Hardy-Weinberg equilibrium. Allele frequency and missingness of each SNP are included in the BOLT-LMM association test output, however, and we recommend checking these values and Hardy-Weinberg p -values (which are easily computed using PLINK2 `--hardy`) when following up on significant associations.

5.6 User-specified filters

Individuals to remove from the analysis may be specified in one or more `--remove` files listing FID and IIDs (one individual per line). Similarly, SNPs to exclude from the analysis may be specified in one or more `--exclude` files listing SNP IDs (typically rs numbers).

6 Association analysis (BOLT-LMM)

6.1 Mixed model association tests

BOLT-LMM computes two association statistics, $\chi^2_{\text{BOLT-LMM}}$ and $\chi^2_{\text{BOLT-LMM-inf}}$, described in detail in our manuscript [1].

- **BOLT-LMM: Association test on residuals from Bayesian modeling using a mixture-of-normals prior on SNP effect sizes.** This approach can fit “non-infinitesimal” traits with loci having moderate to large effects, allowing increased association power.
- **BOLT-LMM-inf: Standard (infinitesimal) mixed model association.** This statistic approximates the standard approach used by existing software.

6.2 BOLT-LMM mixed model association options

The BOLT-LMM software offers the following options for mixed model analysis:

- `--lmm`: Performs default BOLT-LMM analysis, which consists of (1a) estimating heritability parameters, (1b) computing the BOLT-LMM-inf statistic, (2a) estimating Gaussian mixture parameters, and (2b) computing the BOLT-LMM statistic *only if an increase in power is expected*. If BOLT-LMM determines based on cross-validation that the non-infinitesimal model is likely to yield no increase in power, the BOLT-LMM (Bayesian) mixed model statistic is not computed.
- `--lmmInfOnly`: Computes only infinitesimal mixed model association statistics (i.e., steps 1a and 1b).
- `--lmmForceNonInf`: Computes both the BOLT-LMM-inf and BOLT-LMM statistics *regardless of whether or not an increase in power is expected* from the latter.

6.2.1 Restricting SNPs used in the mixed model

If millions of SNPs are available from imputation, we suggest including ≤ 1 million SNPs at a time in the mixed model (using the `--modelSnps` option) when performing association analysis. Using an LD pruned set of ≤ 1 million SNPs should achieve near-optimal power and correction for confounding while reducing computational cost and improving convergence. Note that even when a file of `--modelSnps` is specified, all SNPs in the genotype data are still tested for association; only the random effects in the mixed model are restricted to the `--modelSnps`. Also note that BOLT-LMM automatically performs leave-one-chromosome-out (LOCO) analysis, leaving out SNPs from the chromosome containing the SNP being tested in order to avoid proximal contamination [4, 9].

6.3 Standard linear regression

Setting the `--verboseStats` flag will output standard linear regression chi-square statistics and p -values in additional output columns `CHISQ_LINREG` and `P_LINREG`. Note that unlike mixed model association, linear regression is susceptible to population stratification, so you may wish to include principal components (computed using other software, e.g., PLINK2 or FastPCA [16] in EIGENSOFT v6.0+) as covariates when performing linear regression.

7 Variance components analysis (BOLT-REML)

Using the `--reml` option invokes the BOLT-REML algorithm for estimating heritability parameters and genetic correlations.

7.1 Multiple variance components

To assign SNPs to different variance components, specify a `--modelSnps` file in which each whitespace-delimited line contains a SNP ID (typically an rs number) followed by the name of the variance component to which it belongs.

7.2 Multiple traits

To perform multi-trait variance components analysis, specify multiple `--phenoCol` parameter-value flags (corresponding to different columns in the same `--phenoFile`). BOLT-REML currently only supports multi-trait analysis of traits phenotyped on a single set of individuals, so any individuals with at least one missing phenotype will be ignored. For D traits, BOLT-REML estimates D heritability parameters per variance component and $D(D-1)/2$ correlations per variance component (including the residual variance component).

7.3 Initial variance parameter guesses

To specify a set of variance parameters at which to start REML iteration (which may save time compared to the default procedure used by BOLT-REML if you have good initial guesses), use `--remlGuessStr="string"` with the following format. For each variance component (starting with the residual term, which is automatically named `env/noise`), specify the name of the variance component followed by the initial guess. For instance, a model with two (non-residual) variance components named `vc1` and `vc2` (in the `--modelSnps` file) could have variance parameter guesses specified by:

```
--remlGuessStr="env/noise 0.5 vc1 0.2 vc2 0.3"
```

Note that the sum of the estimates must equal 1; BOLT-REML will automatically normalize the phenotype accordingly.

For multi-trait analysis of D traits, the `--remlGuessStr` needs to specify both guesses of D variance proportions and $D(D-1)/2$ pairwise correlations per variance component. Viewing these values as entries of an upper-triangular matrix (with variance proportions on the diagonal and correlations above the diagonal), you should specify these $D(D+1)/2$ values after each variance component name by reading them off left-to-right, top-to-bottom.

7.4 Trading a little accuracy for speed

BOLT-REML uses a Monte Carlo algorithm to increase REML optimization speed [11]. By default, BOLT-REML performs an initial optimization using 15 Monte Carlo trials and then refines parameter estimates using 100 Monte Carlo trials. If computational cost is a concern (or to perform exploratory analyses), you can skip the refinement step using the `--remlNoRefine` flag. This option typically gives 2–3x speedup at the cost of $\approx 1.03x$ higher standard errors.

8 Output

8.1 BOLT-LMM association test statistics

BOLT-LMM association statistics are output in a tab-delimited `--statsFile` file with the following fields, one line per SNP:

- SNP: rs number or ID string
- CHR: chromosome
- BP: physical (base pair) position
- GENPOS: genetic position either from `bim` file or interpolated from genetic map
- ALLELE1: first allele in `bim` file (usually the minor allele), used as the effect allele
- ALLELE0: second allele in `bim` file, used as the reference allele
- A1FREQ: frequency of first allele
- F_MISS: fraction of individuals with missing genotype at this SNP
- BETA: effect size from BOLT-LMM approximation to infinitesimal mixed model
- SE: standard error of effect size
- P_BOLT_LMM_INF: infinitesimal mixed model association test p -value
- P_BOLT_LMM: non-infinitesimal mixed model association test p -value

Optional additional output. To output chi-square statistics for all association tests, set the `--verboseStats` flag.

8.2 BOLT-REML output and logging

BOLT-REML output (i.e., variance parameter estimates and standard errors) is simply printed to the terminal (`stdout`) when analysis finishes. Both BOLT-LMM and BOLT-REML write output to (`stdout` and `stderr`) as analysis proceeds; we recommend saving this output. If you wish to save this output while simultaneously viewing it on the command line, you may do so using

```
./bolt [... list of options ...] 2>&1 | tee output.log
```

9 Change log

- Version 2.0 (Mar 13, 2015): Added BOLT-REML algorithm for estimating heritability parameters. Fixed parameter-initialization bug that prevented BOLT-LMM from running on some systems. Implemented various minor improvements to parameter-checking.
- (Dec 8, 2014): Licensed source code under GPLv3.
- Version 1.2 (Nov 4, 2014): Added support for testing imputed SNPs in 2-dosage format (RicoPili/plink2 format=2). Fixed bug causing `nan` heritability estimates.
- Version 1.1 (Oct 17, 2014): Added support for testing imputed SNPs with probabilistic dosages.
- Version 1.0 (Aug 8, 2014): Initial release.

10 Website and contact info

Software updates will be posted at the following website:

`http://www.hsph.harvard.edu/alkes-price/software/`

If you have comments or questions about the BOLT-LMM software, please contact Po-Ru Loh, `loh@hsph.harvard.edu`.

11 License

- The BOLT-LMM source code is free software under the GNU General Public License v3.0 (GPLv3).
- The BOLT-LMM executable is freely available but not open source because it was built with the proprietary Intel Math Kernel Library (under a non-commercial license). Specific license terms are as follows.

Copyright ©2014 Harvard University. All rights reserved. This software is supplied without any warranty or guaranteed support whatsoever. Harvard University cannot be responsible for its use, misuse, or functionality. The software may be freely copied for non-commercial purposes provided this copyright notice is retained.

Starting from v2.0, the BOLT-REML component of the software also uses routines from the NLOpt library written by Steven G. Johnson and distributed under the MIT License (Copyright ©2007-2011 Massachusetts Institute of Technology).

References

1. Loh, P.-R. *et al.* Efficient Bayesian mixed model analysis increases association power in large cohorts. *Nature Genetics* (2015).
2. Kang, H. M. *et al.* Variance component model to account for sample structure in genome-wide association studies. *Nature Genetics* **42**, 348–354 (2010).
3. Lippert, C. *et al.* FaST linear mixed models for genome-wide association studies. *Nature Methods* **8**, 833–835 (2011).
4. Listgarten, J. *et al.* Improved linear mixed models for genome-wide association studies. *Nature Methods* **9**, 525–526 (2012).
5. Listgarten, J., Lippert, C. & Heckerman, D. FaST-LMM-Select for addressing confounding from spatial structure and rare variants. *Nature Genetics* **45**, 470–471 (2013).
6. Lippert, C. *et al.* The benefits of selecting phenotype-specific variants for applications of mixed models in genomics. *Scientific Reports* **3** (2013).
7. Zhou, X. & Stephens, M. Genome-wide efficient mixed-model analysis for association studies. *Nature Genetics* **44**, 821–824 (2012).
8. Svishcheva, G. R., Axenovich, T. I., Belonogova, N. M., van Duijn, C. M. & Aulchenko, Y. S. Rapid variance components-based method for whole-genome association analysis. *Nature Genetics* (2012).
9. Yang, J., Zaitlen, N. A., Goddard, M. E., Visscher, P. M. & Price, A. L. Advantages and pitfalls in the application of mixed-model association methods. *Nature Genetics* **46**, 100–106 (2014).
10. Yang, J., Lee, S. H., Goddard, M. E. & Visscher, P. M. GCTA: a tool for genome-wide complex trait analysis. *American Journal of Human Genetics* **88**, 76–82 (2011).
11. P-R, L. *et al.* Contrasting regional architectures of schizophrenia and other complex diseases using fast variance components analysis. *bioRxiv* (2015).
12. Johnson, S. G. The NLOpt nonlinear-optimization package. URL <http://ab-initio.mit.edu/nlopt>.
13. Purcell, S. *et al.* PLINK: a tool set for whole-genome association and population-based linkage analyses. *American Journal of Human Genetics* **81**, 559–575 (2007).
14. Chang, C. C. *et al.* Second-generation PLINK: rising to the challenge of larger and richer datasets. *GigaScience* (2015).
15. Howie, B. N., Donnelly, P. & Marchini, J. A flexible and accurate genotype imputation method for the next generation of genome-wide association studies. *PLoS Genetics* **5**, e1000529 (2009).

16. Galinsky, K. J. *et al.* Fast PCA of very large samples in linear time. Abstract presented at the 64th Annual Meeting of The American Society of Human Genetics, October 18–22, 2014, San Diego, CA.