MONSTER Software Documentation

Version 1.3
December 17, 2016

Duo Jiang[1], Mary Sara McPeek[2,3]

Department of Statistics[1]
Oregon State University, Corvallis, OR 97330, USA

Departments of Statistics[2] and Human Genetics[3]
The University of Chicago, Chicago, IL 60637, USA

MONSTER
A C++ program for robust SNP-set association testing for quantitative traits in samples with related individuals

Homepage: http://www.stat.uchicago.edu/~mcpeek/software/index.html
Release 1.3 December 17, 2016

We request that use of this software be cited in publications as follows:
Jiang D., McPeek M. S. (2014). Robust Rare Variant Association Testing for Quantitative Traits in Samples with Related Individuals. Genetic Epidemiology 38(1):10-20

Contents

1 Overview of MONSTER

MONSTER is a C++ program that performs a joint test for association between a set of variants and a quantitative trait in samples with related individuals. Any combination of rare and common variants may be included in the joint test, and the variants may come from, e.g., sequence or genotype data. In particular, MONSTER is suitable for testing for association between a trait and a set of rare variants. MONSTER performs a rapid and robust test against the null hypothesis that none of the variants in the set is associated with the phenotype.

MONSTER is applicable to completely general combinations of related and unrelated individuals, provided that the kinship coefficients are known or can be estimated. For example, it is equally applicable to complex inbred pedigrees and to simpler study designs consisting of unrelated individuals and small outbred families. The program allows the possibility that the kinship coefficients are unknown but can be estimated, for example, based on genome-wide data.

The MONSTER testing method uses a mixed effects model that accounts for covariates and additive polygenic effects for the phenotype, and adaptively adjusts to the unknown configuration of the effects of the tested SNPs to achieve robust performance across a wide range of possible genetic architecture of the trait. It can be viewed as a generalization of the SKAT-O method [4] to allow relatedness among sampled individuals. Specifically, the MONSTER test statistic is equivalent to a convex combination between the test statistics of two existing methods, famBT [2] and famSKAT [2], which are derived based on different modeling assumptions on the genetic architecture and are powerful in different scenarios. The relative weights of the two methods are adaptively determined by the data to best fit the unknown genetic architecture. The resulting method, MONSTER, tends to either mimic the performance of the better-performing of famSKAT and famBT or to outperform both methods. This robustness of power to the underlying modeling assumptions can be very desirable in practice, as the true biological mechanism and genetic architecture is usually unknown *a priori* and can be expected to vary across genes and traits. More detail on the MONSTER method can be found in the MONSTER paper, Jiang and McPeek (2014)[1].

In addition to the p-value for the MONSTER method, the program can, at the user's option, also output the p-values for famBT and famSKAT. This feature is useful for comparing results from the three tests. How the p-values of the three methods compare in each specific scenario may depend on the usually unknown information on the genetic architecture of the trait.

The program also allows the option of fitting a linear mixed effects model to the phenotype and covariate data without performing any association test. This can be useful for preliminary analyses of the phenotype and covariate data in order to formulate the null model.

2 Installing MONSTER

2.1 Installation Prerequisites

Successful installation of MONSTER requires The GNU Scientific Library. It is free software under GNU GPL that can obtained at `http://www.gnu.org/software/gsl/`, in case it is not available on your system. It is also assumed that a GNU g++ compiler is available on your system.

2.2 Installation Instructions

1. Download the MONSTER package. This package contains the documentation, source code, example files, and the GNU GPL license.

2. Read the entire documentation (this document) carefully to understand the purpose of this program and how it works. It will also be helpful to read the paper on the MONSTER method, Jiang and McPeek (2014)[1].

3. Uncompress the archive: `tar xvfz MONSTER_v1.3.tar.gz`

4. Switch to the newly created directory: `cd MONSTER/`

5. This directory contains the GNU GPL license in file `gpl.txt` and three subdirectories:

   - `src` contains the source code;
   - `doc` contains this document `MONSTER_v1.3_doc.pdf`;
   - `examples` contains example input and output files.

6. Switch to the `src` directory: `cd src/`

7. Type `make`. This will build an executable program called MONSTER.

3 Running MONSTER

MONSTER is run from the command line via the command `./MONSTER` with all information specified by command line options. To run the executable program (see Section 2), first prepare the input files (see Section 4). Then, to run MONSTER using the default filenames and options, simply type on the command line:

```
./MONSTER
```

Alternatively, to change the input filenames or to use other options, use flags in the command line. The following three examples are what the command might look like:

```
./MONSTER -p pheno.txt -g geno.txt -s SNP.txt -k kin.txt -c -r
./MONSTER -g chr14.txt -k kinship.txt -m 1.0 -E -e eigfilename
./MONSTER -p pheno.txt -k kin.txt -n -e eigfilename
```

We briefly summarize the usage of the available flags below. More details about input file specifications can be found in Section 4.

`-p pheno.txt` Allows the user to specify the name of the phenotype data input file. This file also includes pedigree information and covariate data. The filename defaults to `pheno.txt` if this flag is not used. To specify another filename, replace `pheno.txt` with the appropriate filename.

`-g geno.txt` Allows the user to specify the name of the genotype data input file. The filename defaults to `geno.txt`. To specify another filename, replace `geno.txt` with the appropriate filename.

-s SNP.txt Allows the user to specify the name of the SNP list input file. This file informs the program which SNPs to include in the tests and offers the opportunity to assign weights to the SNPs. The filename defaults to `SNP.txt`. To specify another filename, replace `SNP.txt` with the appropriate filename.

-k kin.txt Allows the user to specify the name of the kinship coefficient input file, which contains the kinship and inbreeding coefficients for all possible pairs of individuals within each family. The filename defaults to `kin.txt`. To specify another filename, replace `kin.txt` with the appropriate filename.

Additional flags

-c Use this flag to instruct the program to perform two other association tests, famBT and famSKAT, in addtion to MONSTER.

-r If this flag is used, MONSTER will output a text file `MONSTER.resid` that gives the phenotypic residuals obtained from the linear mixed effects model under the null hypothesis of no association. Both standardized and unstandardized residuals will be included. Details can be found in Section 5.3.

-m 0 This flag sets a threshold for the fraction of SNPs in the SNP set that can be missing for any individual to be included in the test. For each SNP set, any individual for whom the fraction of tested SNPs that have missing genotypes is beyond (not including) the threshold will be excluded from the test. For example, `-m 0.1` instructs the program to consider in each test only the individuals for whom at least 90% of the SNPs in the tested SNP set have non-missing genotypes. The threshold should be specified after `-m` as a decimal number between 0 and 1 inclusive (not as a percentage value). If this option is not used, the program will use 0 as the default threshold and include in the test only the individuals with no missing genotypes. In order for the program to include all individuals regardless of missingness, use `-m 1`. An error will be reported if the threshold is outside of the interval [0,1]. The program will not remove any SNP in the SNP list file from the analysis. However, a warning message will be given if a tested SNP has missing genotypes for over 50% of the individuals.

-E or **-A** This flag customizes the imputation scheme to use for the missing genotypes after removing individuals for whom the fraction of missing SNPs is beyond the threshold. The program offers three possible imputation methods: the best linear unbiased predictor (BLUP) [5], the best linear unbiased estimator (BLUE) [3], or the sample average over all the non-missing genotypes for a particular SNP. A major factor to be considered when choosing an imputation method is computation time. BLUP is generally the slowest among the three methods, with sample average being the fastest, and the difference could be substantial when the sample includes large families. For detailed suggestions on how to choose between the three methods, please refer to Section 7.3. Without this option, BLUP will be used as the default. `-E` and `-A` instruct the program to use BLUE and the sample average, respectively; at most one of `-E` and `-A` can be used, and an error message will be returned if otherwise. We note that this option is relevant only when there is indeed missingness in the genotype data and meanwhile the missingness threshold set by `-m` is above 0.

-n This flag instructs the program to only fit the linear mixed effects model under the null hypothesis, without performing any association test. If it is used, no genotype data file or SNP list file will be read and all individuals will be included when fitting the null model. Flag `-r` can be used in conjunction with this option in order for the program to output the phenotypic residuals, in addition to the parameter estimation results. However, with the use of `-n`, flags `-m` and `-A` (or `-E`) will be ignored.

-e eigfilename Allows the user to indicate the availability of a file containing eigen-decomposition results for the kinship matrix and to specify its filename. There is no default filename. This option can only be used if the value in flag `-m` equals 1 or if flag `-n` is used (in either case, all individuals will be included in the analysis). It instructs the program to improve computational efficiency by making use of existing eigen-decomposition results stored in the file under the name `-eigfilename` (can be replaced by another filename). When this option is not used but the missingness threshold is set to be 1, the program will perform the eigen-decomposition only once and use the results for all of the SNP sets. This option will be ignored if it is used with a missingness threshold strictly smaller than 1.

To illustrate the meanings of the flags, we explain what the following three example commands do. More examples are provided in Section 6.

- `./MONSTER -p pheno.txt -g geno.txt -s SNP.txt -k kin.txt -c -r`

  This command instructs the program to read data from the *.txt files. Due to flag `-c`, famBT and famSKAT will be performed, in addition to the MONSTER method. WIth the `-r` flag, the program will output phenotypic residuals in the file `MONSTER.resid`. Since no `-m` is given, the missingness threshold defaults to zero and thus individuals with any missing genotypes will be removed from the analysis.

- `./MONSTER -g chr14 -k kinship -m 1.0 -E -e eigresult`

  This command instructs the program to read genotype data and kinship coefficients from the files `chr14` and `kinship`, respectively. Since no `-p` or `-s` is given, the program will use the default filenames `pheno.txt` and `SNP.txt` for the phenotype data file and the SNP list file. The missingness threshold is set to be 1, so all individuals will be included in the analysis. The flag `-e eigresult` informs the program that eigen-decomposition results are already available and should be found in the file `eigresult`. With the `-E` flag, BLUE imputation will be used. Only the MONSTER testing method will be performed in absence of `-c`.

- `./MONSTER -p pheno.txt -k kin.txt -n -e eigresult`

  This command instructs the program to only estimate the MLEs under the null hypothesis without running any hypothesis test and to output the results in `MONSTER.param`. All individuals will be included in the analysis. Phenotype data and kinship coefficients are read from `pheno.txt` and `kin.txt`, respectively. No genotype data file or SNP list file will be used. The program will read eigen-decomposition results from the file `eigresult`. There will be two

4 Input

1. **Pedigree and phenotype data file (specified by flag -p)**

   The pedigree and phenotype data file contains the pedigree structure, values of the quantitative phenotype and any covariates. The columns in the file are:

   ```
   family ID (positive integer)
   individual ID (positive integer)
   father's ID (integer, not used by the program)
   mother's ID (integer, not used by the program)
   sex (1=male, 2=female)
   phenotype (numeric)
   covariate1 (numeric)
   covariate2 (numeric)
       ⋮
   ```

   The requirements are the following:

   - Tab or space delimited.
   - Only integers are allowed for the individual and family IDs in the current version. The individual ID is assumed to be *unique* across all pedigrees. Currently, the program will not check whether the IDs follow the correct format.
   - Individuals in the same family must be grouped together in the file (order within families does not matter).
   - Sampled individuals who are unrelated to anyone else in the sample should be included in this file by giving each such person their own unique family ID.
   - Intercept should NOT be included in this file as a covariate. When the program performs the test, it will automatically add an intercept to the phenotype model. If an intercept is also input as a covariate, the program will likely terminate abnormally before yielding any result.
   - **When an empirical kinship matrix is used** for the analysis, the pedigree and phenotype data file should list the **same family ID** for all individuals in the analysis. (If different family IDs are used, the program will assume a block diagonal structure for the kinship matrix.)

   Example
   A file with 2 pedigrees with 4 individuals each, one phenotype, and 2 covariates might look like:

   ```
   3 1 0 0 1 1.2 1.3 1.4
   3 2 0 0 2 2.2 2.3 2.4
   3 3 1 2 2 3.3 3.4 7.00
   3 4 1 2 2 4.4 4.4 5.5
   2 5 0 0 1 1.2 1.3 1.4
   2 6 0 0 2 2.2 2.3 2.4
   2 7 5 6 2 3.3 3.4 -12
   2 9 5 6 2 4.4 4.4 5.5
   ```

The optional flag `-p pheno.txt` allows the user to specify the name of the pedigree and phenotype data file. The filename defaults to `pheno.txt` if this flag is not used. To specify another filename, replace `pheno.txt` with the appropriate filename.

2. **Genotype data file (specified by flag -g)**

The genotype data file stores each individual's genotypes. Its rows refer to the SNPs and its columns refer to the individuals. All the SNPs to be included in the association tests should be present, but the file is allowed to contain additional SNPs that are not in any test. The requirements are the following:

- Tab or space delimited.
- The first row is a header line that starts with a '0', followed by the individual IDs. Every person in this file has to be present in the pedigree and phenotype data file and vice versa. Individuals must be listed in the same order as in the pedigree and phenotype data file. If not, an error will be reported.
- The first column lists the unique IDs of the SNPs. Numeric and alphanumeric IDs are allowed.
- Genotypes should be coded by the number (0, 1 or 2) of copies of the tested variant (for example, the minor allele) an individual has at the corresponding site. To account for imputed genotypes (for example, genotype dosage or BLUP [5]), the program allows for non-integer genotype values. Missing genotypes should be coded by -9.
- No SNP included in the data file should be missing on all the individuals. The program will return an error if such a SNP is found. Any SNP for which over 50% of the individuals have missing genotypes will be reported with a warning message.
- In principle, the genotypes for a bi-allele site can be coded by the number of copies of either the minor allele or the major allele. However, the two ways of coding are NOT interchangeable in the test, and correspond to different modeling assumptions on the genetic effects of the variants on the phenotype. Specifically, the effects of the alleles used for genotype coding are assumed to be similar and correlated across SNPs in a tested SNP set (see Section 4.3). Different coding choices will in general lead to different testing results. Prior functionality knowledge on the SNPs may be helpful in making the choice. In absence of prior information, we suggest the use of the minor allele for genotype coding, particularly for rare-variant sites, because rare alleles are more likely to be causal and detrimental according to standard evolutionary theory.
- A genetic site with $K \geq 3$ alleles (or variants) may be coded in $K-1$ rows. To do this, the user should first select $K-1$ alleles to be the "coding alleles", and specify in k^{th} the number of copies of the k^{th} coding allele, $1 \leq k \leq K-1$. Similar to the bi-allele case, it does matter to the test which of the K alleles are chosen to be the coding alleles. Different choices correspond to different modeling assumptions on the genetic effects of the alleles on the phenotype, and will lead to different testing results. Specifically, it is assumed that the effects of the $K-1$ coding alleles are similar and correlated among each other and with the effects of the coding alleles of the other sites in a tested SNP set. See details on the modeling assumptions in Jiang and McPeek (2014) [1].
- For a dataset with 3 individuals and 4 markers, the genotype file might look like

```
0         2     3     5
SNP1      0     1.42  1
Site2     2     -9    0
rs424964  1     2     0
920       0.89  2.1   1
```

The optional flag `-g geno.txt` allows the user to specify the name of the genotype data file. The filename defaults to `geno.txt` if this flag is not used. To specify another filename, replace `geno.txt` with the appropriate filename.

3. **SNP list file (specified by flag -s)**

The SNP list file informs the program about the IDs of the SNPs in each of the SNP sets to be tested, and it offers the opportunity for the user to specify known weights for the effect size of each of the SNPs. Each SNP set must contain at least 2 SNPs. A SNP set is specified by one or two lines in the SNP list file: The first line, the SNP set info line, is always required; A second line, the SNP weight line, should follow the first line when the user would like to specify weights for the SNPs. If no SNP weight line is provided for a SNP set, the program will use equal weights for all the SNPs in the set.

For each SNP set, the SNP info line must have the following columns:

```
SNP set ID (alphanumeric)
whether weights will be specified (0=not specified, 1=specified)
ID of SNP 1 (alphanumeric)
ID of SNP 2 (alphanumeric)
```

⋮

If the second column in the SNP set info line is 1, there must be a SNP weight line immediately after. The SNP weight line should have the following columns.

```
SNP set ID (must be identical to the previous line)
0
weight for SNP 1
weight for SNP 2
```

⋮

Additional requirements on the file are:

- Tab or space delimited.
- Only alphanumeric IDs are allowed and they should match exactly with those in the genotype data file. All SNPs in the file should be present in the genotype data file. An error will be reported if otherwise.
- The ordering of the SNP sets or that of the SNPs within each set does not matter.
- The maximum number of SNPs that can included in a SNP set is set to be 1000 by the current version. To modify this number, open `MONSTER.cpp` in the directory `MONSTER/src` and replace `1000` in `#define MAXSNPSET 1000` by the appropriate upper limit.

- A SNP weight line must contain the same number of entries as the preceding SNP set info line.
- Weights must be non-negative numbers. Negative weights will result in an error. At least one SNP in each SNP set needs to have a strictly positive weight. Otherwise, the SNP set will not be analyzed.
- At least two SNPs with strictly positive weights should be included in each SNP test. If a SNP set does not satisfy this condition, the program will return an warning message and will not perform the MONSTER test for this set.
- Below is what a SNP list file might look like, if there are 4 SNP sets, only the second of which have specified weights.

```
SNPset2  0    SNP3      SNP10      SNP9
 gene1   1    SNP21      SNP1      SNP4      SNP8
 gene1   0      1        3.21       2.2       0.2
 BRCA1   0  rs1799950  rs4986850  rs2227945  rs16942  rs1799966
SNPset1  0  variant7   variant20
```

The optional flag `-s SNP.txt` allows the user to specify the name of the SNP list input file. The filename defaults to `SNP.txt` if this flag is not used. To specify another filename, replace `SNP.txt` with the appropriate filename.

4. **Kinship coefficient file (specified by flag -k)**

The kinship coefficient file contains the pairwise kinship coefficients between individuals and the inbreeding coefficient for each individual. The columns in the file should be

```
familyID    indiv_1ID    indiv_2ID    coefficient
```

If `indiv_2ID` $\neq$ `indiv_2ID`, then `coefficient` is the kinship coefficient between the two individuals. If `indiv_1ID = indiv_2ID`, then `coefficient` is the individual's **inbreeding coefficient** (not the self-kinship coefficient, so if the individual is outbred, this value should be 0, not 1).

The requirements on the file are the following:

- Tab or space delimited.
- A kinship coefficient should be provided for every pair of individuals within each family in the pedigree and phenotype data file. An inbreeding coefficient should be provided for every individual (even if the person is outbred). A sampled individual who does not share a family ID with anyone else should be represented in the kinship coefficient file by a single line that specifies the individual's inbreeding coefficient value. The program will NOT check if this is the case or not.
- The file must contain inbreeding coefficients, h_i, instead of self-kinship coefficients, Φ_{ii}, where $\Phi_{ii} = \frac{1}{2} + \frac{1}{2}h_i$ and $h_i = 2\Phi_{ii} - 1$. Thus, for an outbred individual, the inbreeding coefficient should be 0, not 1. An inbreeding coefficient higher than 0.1 is alarming for most human samples, and, if such a coefficient is found in the kinship coefficient file, the program will report a warning message.

- Individuals in the same family should be grouped together in the file. Family and individual IDs should match exactly with those in the pedigree and phenotype data file. The program will NOT check if this is the case or not.
- Families must be listed in the same order as in the pedigree and phenotype data file. Failure to do so will likely result in a segmentation fault.
- It is assumed that everyone in the pedigree and phenotype data file is also in the kinship file, and no one in the kinship file is not in the pedigree and phenotype data file.
- To calculate the pedigree-based kinship and inbreeding coefficients, one could use the `KinInCoef` software, downloadable at http://galton.uchicago.edu/~mcpeek/software/KinInbcoef/index.html The output file of this program has the exact format required for the MONSTER kinship coefficient input file.
- The MONSTER program does not require the input kinship and inbreeding coefficients to be calculated based on the pedigree structure. One may choose to use empirical kinship and inbreeding coefficients instead. Just be careful to input the inbreeding coefficients, $h_i = 2\Phi_{ii} - 1$, not the self-kinship coefficients, Φ_{ii}, when `indiv1ID = indiv2ID`.

The optional flag `-k kin.txt` allows the user to specify the name of the kinship coefficient file. The filename defaults to `kin.txt` if this flag is not used. To specify another filename, replace `kin.txt` with the appropriate filename.

5. **Optional eigen-decomposition file (specified by flag -e)**

The eigen-decomposition file is optional. Is only relevant when the missingness threshold set by flag `-m` equals 1 or when flag `-n` is used. It allows the program to improve computational efficiency by making use of existing eigen-decomposition results of the kinship matrix. It is a binary file that encodes the eigenvalues and eigenvectors of the kinship matrix in `double` (double precision floating-point type). Let Φ be the $n \times n$ kinship matrix, whose (i, j)th element is $2\Phi_{ij}$, where Φ_{ij} is the kinship coefficient between individuals i and j for $i \neq j$, and where $2\Phi_{ii} = 1 + h_i$, where h_i is the inbreeding coefficient of individual i. Let $\Phi = VDV^{-1}$ be an eigen-decomposition of Φ, where D is a diagonal matrix with the eigenvalues as the diagonal elements and V is an orthogonal matrix containing the eigenvectors. The eigen-decomposition input file should encode the diagonal elements of D followed by a row-wise list of the elements of V in binary format as double precision floating-point numbers.

- The kinship matrix Φ underlying these files should correspond exactly to the set and ordering of individuals in the pedigree and phenotype data file. In particular, the dimension of Φ should equal the number of individuals in the pedigree and phenotype data file. Mismatched dimension will likely result in a segmentation fault.
- A scenario in which eigen-decomposition results may be available prior to the analysis occurs when, prior to the current analysis, MONSTER has been run on the same set of individuals (with the same kinship coefficients) as in the current analysis, and thus the eigen-decomposition results can be reused.

The optional flag `-e eigfilename` allows the user to specify the name of the eigen-decomposition file. There is no default filename. The option can only be used when the missingness threshold set by flag `-m` equals 1 or when flag `-n` is used. It instructs the program to read existing eigenvalue and eigenvector results off the file eigfilename without performing the decomposition. To specify another filename, replace `eigfilename` by the appropriate filename.

5 Output

The program will output up to 4 files: a text file recording the primary results of the association analysis, another text file that contains the parameter estimation results, an optional file that lists two types of phenotypic residuals under the null hypothesis, and another optional binary file that records the eigenvalues and the eigenvectors of the kinship matrix. The file for the results of association analysis will not be part of the output if flag `-n` is used.

1. **MONSTER.out** is the primary output file for the results of the association tests. This file lists the testing result for each of SNP sets in the order they are included in the SNP list file. It contains the following information:

   - ID of the SNP set.
   - The total number of individuals and the number of SNPs in the test.
   - The value of the optimal ρ selected by the MONSTER method.
   - The p-value returned by the MONSTER method and, if the `-c` flag is used, the p-values of the famBT and famSKAT methods in addition. See the MONSTER paper, Jiang and McPeek (2014) [1], for details on the three tests.

2. **MONSTER.param** is a text file containing variance component and covariate parameter estimates under the null hypothesis of no association. Each SNP set corresponds to its own list of parameter estimation results. With 2 SNP sets, the file might look like:

```
SNPset1
Parameter     nullMLE     SE_nullMLE
Heritability  0.512986    0.003957
Additive_Var  6.626302    1.076178
Error_Var     6.290817    0.526706
Intercept     2.323301    0.198892
Covariate_1   0.892910    0.078998
Covariate_2   1.902201    0.725178

SNPset2
Parameter     nullMLE     SE_nullMLE
Heritability  0.520929    0.004011
Additive_Var  6.720991    1.107038
Error_Var     6.180942    0.558902
Intercept     2.190042    0.208411
Covariate_1   0.908924    0.089284
Covariate_2   1.880983    0.789920
```

3. **MONSTER.resid** is a text file giving phenotypic residuals for all the tested individuals obtained from the linear mixed model under the null hypothesis of no association. It will be part of the output only when flag `-r` is used. Two types of residuals are included, both could be useful for model diagnostics purposes. Each SNP set corresponds to its own set of residuals presented in four consecutive lines in the file:

   - The first line gives the ID of the SNP set, or "NullModel" if flag `-n` is used.
   - The second line lists the IDs of the n individuals included in the test.
   - The third line lists the n untransformed residuals, $r_i = y_i - \hat{y}_i$, $i = 1, \cdots, n$, where y_i is the observed phenotypic value of the i^{th} individual in the second line and $\hat{y}_i$ is the fitted phenotypic value based on the model under the null hypothesis. Keep in mind that, due to relatedness among the sampled individuals, the untransformed residuals may not have the same properties (for example, exchangeable and orthogonal with the covariates) as the residuals in an ordinary linear regression.
   - The fourth line lists the n transformed residuals defined as $C^{-T}r$, where $r = (r_1, \cdots, r_n)^T$ is a vector for the untransformed residuals, C is an $n \times n$ matrix such that $C^T C = \hat{\Sigma}$ and $\hat{\Sigma}$ is the estimated phenotypic covariance matrix under the null hypothesis of no association.

   With two SNP sets, the residual output file might look like the following:

   ```
   SNPset1
   1 2 3 5 6 7
   0.21 -0.19 1.72 0.08 -2.04 -0.11
   0.19 -0.33 1.14 -0.03 -1.24 0.27

   SNPset2
   1 3 4 6
   0.18 0.79 0.32 -1.96
   0.13 1.02 0.30 -1.45
   ```

4. **MONSTER.eig** is a binary file that stores the eigenvalues and eigenvectors of the kinship matrix, as computed by the MONSTER program. It will be part of the output only when the missingness threshold specified by flag `-m` equals 1 and meanwhile the flag `-e` is not used, or when flag `-n` is used. The formatting of this output file is the same as that of the binary eigen-decomposition input file described in Section 4.5.

6 Examples

The directory `MONSTER/examples` provides example input files: `pheno_ex.txt`, `geno_ex.txt`, `SNP_ex.txt`, `kin_ex.txt` and `eigfile_ex`. Below, we list several example commands that can be run on these files.

1. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt`

   This command instructs the program to perform only the MONSTER testing method, not famBT or famSKAT. Without the flag `-m`, individuals with any missing genotypes will be removed from the analysis. Eigen-decomposition of the kinship matrix is done for each SNP set. The program will generate two output files: `MONSTER.out` and `MONSTER.param`.

An equivalent command is
`./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -m 0`

2. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -c -r`

   This command will generate three output files: `MONSTER.out`, `MONSTER.param` and `MONSTER.resid`. With flag `-c`, `MONSTER.out` will contain results of the famBT and famSKAT methods, in addition to the result of MONSTER. Eigen-decomposition of the kinship matrix is redone for each of the SNP sets.

3. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -m 0.1`

   This command instructs the program to perform only the MONSTER method, not famBT or famSKAT. The missingness threshold is set to be 0.1, and thus individuals whose genotypes are missing for over 10% of the SNPs in the tested SNP set will be removed from the test. As the default, BLUP imputation will be used for the remaining missing genotypes. Eigen-decomposition of the kinship matrix is redone for each of the SNP sets. The program will generate two output files: `MONSTER.out` and `MONSTER.param`.

4. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -m 0.333 -E`

   This command instructs the program to perform only the MONSTER method, not famBT or famSKAT. Since the missingness threshold is set to be 0.333, individuals whose genotypes are missing for over 33.3% of the SNPs in the tested SNP set will be removed from the test. BLUE will be used to impute any remaining missing genotypes. Eigen-decomposition of the kinship matrix is redone for each of the SNP sets. The program will generate two output files: `MONSTER.out` and `MONSTER.param`.

5. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -m 1 -A`

   This command instructs the program to perform only the MONSTER method, not famBT or famSKAT. Since the missingness threshold is set to be 1, all individuals will be included in the analysis. Sample averages will be used to impute missing genotypes. Eigen-decomposition of the kinship matrix is done only once with the results reused for all SNP sets, and these results will be output in `MONSTER.eig`. The program will generate three output files: `MONSTER.out`, `MONSTER.param` and `MONSTER.eig`.

6. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -m 1 -e eigfile_ex`

   This command instructs the program to perform only the MONSTER method, not famBT or famSKAT. Since the missingness threshold is set to be 1, all individuals will be included in the analysis regardless of missingness. BLUP imputation will be used for missing genotypes. The program will not perfrom eigen-decomposition for the kinship matrix, and will instead read the already available results from the file `eigfile_ex`. The program will generate two output files: `MONSTER.out` and `MONSTER.param`.

7. `./MONSTER -p pheno_ex.txt -k kin_ex.txt -n`

   This command instructs the program to only estimate the MLEs under the null hypothesis without running any hypothesis test. Phenotype data and kinship coefficients are read from `pheno_ex.txt` and `kin_ex.txt`, respectively. All individuals will be included in the analysis. No genotype data file or SNP list file will be read. The program will generate two output files: `MONSTER.param` and `MONSTER.eig`.

8. `./MONSTER -p pheno_ex.txt -k kin_ex.txt -n -r`

   This command instructs the program to only fit the linear mixed effects model under the null hypothesis without running any hypothesis test. Phenotype data and kinship coefficients are read from `pheno_ex.txt` and `kin_ex.txt`, respectively. All individuals will be included in the analysis. No genotype data file or SNP list file will be read. The parameter estimation results will be output in `MONSTER.param`, the phenotypic residuals in `MONSTER.resid`, and the eigen-decomposition results in `MONSTER.eig`.

Examples of inappropriate usage:

1. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -m 0.25 -e eigfile_ex`

   This is an inappropriate command, because flag `-e` should not be used when the missingness threshold is below 1. As a result, the option `-e eigfile_ex` will be ignored.

2. `./MONSTER -p pheno_ex.txt -g geno_ex.txt -s SNP_ex.txt -k kin_ex.txt -A -E`

   This is an inappropriate command, because `-A` and `-E` may not be used simultaneously. An error will be reported.

7 FAQ

1. **Why is the method named MONSTER?**
   MONSTER is short for Minimum p-value Optimized Nuisance parameter Score Test Extended to Relatives. Please refer to Jiang and Mcpeek (2014) [1] for details.

2. **Do I need to assess the fit of the null model before running the test?**
   Yes, it is highly recommended that the user do preliminary analyses on the phenotype and covariate data to formulate the null model, before deciding to apply it in the association tests. In order to fit a null model without performing any association test, use flag `-n`. Some output files provided by the program may be helpful for evaluating the null model. For example, the two types of residuals in `MONSTER.resid` and the parameter estimation results in `MONSTER.param` may be relevant for model diagnostics purposes and for choosing covariates as well as possible transformations of the phenotypes and/or covariates.

3. **How should I choose between the imputation schemes for the missing genotypes?**
   The program offers three possible ways to impute missing genotypes: BLUP (the default), BLUE (specified by `-E`) or the sample average (specified by `-A`). Both BLUP and BLUE take advantage of the family structure among the related individuals. BLUP is taken as the default method because it tends to be more accurate and informative than BLUE. However, BLUP imputation can be slow, if the family sizes are quite large (hundreds of individuals in

one family, for example) and meanwhile there are many SNPs (tens of variants, for example) to be imputed. When BLUP imputation turns out to be too slow or is expected to be slow, we recommend using BLUE as a faster alternative. The program will give a warning message if BLUP is used and the largest family size exceeds 200. If the sample contains unrelated individuals and the kinship coefficients are not pedigree-based, we recommend the use of sample average imputation.

4. **What if the pedigree structure is not available?**
   The MONSTER testing method depends on the pedigree structure only through the input kinship and inbreeding coefficients. The father IDs and mother IDs in the pedigree and phenotype data file are not used by the program, and thus their values can be arbitrarily assigned for the sake of this program only. Just be careful that, when the kinship input file is based on empirical kinship and inbreeding coefficients estimated for the whole sample, all individuals must be assigned the same family ID in the pedigree file and the kinship file.

8 Acknowledgments

We gratefully acknowledge:

- Eigen. We used the package for some of the matrix computations.
- The CompQuadForm package. We used code from it for approximating the distribution function of a linear combination of independent chi-squared random variables.
- R. We used the Brent_fmin subprogram from the function optimize().

9 References

[1] Duo Jiang and Mary Sara McPeek. 2014. Robust Rare Variant Association Testing for Quantitative Traits in Samples with Related Individuals. Genetic Epidemiology 38:10-20

[2] Han Chen, James B. Meigs and Josée Dupuis. 2013. Sequence Kernel Association Test for Quantitative Traits in Family Samples. Genet Epidemiology 37:196-204

[3] Mary Sara McPeek, Xiaodong Wu and Carole Ober. 2004. Best Linear Unbiased Allele-Frequency Estimation in Complex Pedigrees. Biometrics 60:359-367

[4] Seunggeun Lee, Michael C. Wu and Xihong Lin. 2012. Optimal Tests for Rare Variant Effects in Sequencing Association Studies. Biostatistics 13:762-775

[5] Mary Sara McPeek. 2012. BLUP Genotype Imputation for Case-Control Association Testing with Related Individuals and Missing Data. Journal of Computational Biology 19:756-765