Journal of Statistical Software

June 2020, Volume 94, Issue 5. doi: 10.18637/jss.v094.i05

AssocTests: An R Package for Genetic Association Studies

Lin Wang
Capital University of Economics and Business

Wei Zhang
Chinese Academy of Sciences

Qizhai Li
Chinese Academy of Sciences

Abstract

The R package **AssocTests** provides some procedures which are commonly used in genetic association studies. These procedures are population stratification correction through eigenvectors, principal coordinates of clusterings, Tracy-Widom test, distance regression, single-marker test, maximum test based on three Cochran-Armitage trend tests, non-parametric trend test, and non-parametric maximum test. The trait values for these methods should be discrete or continuous. The discrete traits can be coded by 1/0 for cases/controls. The genotype values can be 0, 1, or 2 indicating the number of risk alleles for a biallelic single-nucleotide polymorphism. This article introduces the methods and algorithms implemented in the package. Some examples are provided to illustrate the package's capability.

Keywords: distance regression, genetic association studies, population stratification, Tracy-Widom test, R.

1. Introduction

Genetic association study has become a popular tool to identify the genetic variants predisposing to human complex diseases (Klein *et al.* 2005; Sladek *et al.* 2007; Burton *et al.* 2007; Ardlie *et al.* 2015). Currently, more than 10,000 deleterious single-nucleotide polymorphisms (SNPs) have been identified (`http://www.genome.gov/gwastudies/`). Two important issues are often considered in population-based genetic association studies. One issue is the adjustment for confounders in which the population stratification (PS) is noteworthy, and the other is the adoption of powerful tests. Single-marker analysis is crucial in many gene- or pathway-based procedures, such as the truncated p value combination method (Zaykin, Zhivotovsky, Westfall, and Weir 2002; Yu *et al.* 2009) and the minimum p value approach (Hoh, Wille, and Ott 2001; Dudbridge and Koeleman 2004). In the single-marker analysis, investigators have

developed many methods to consider the genetic mode of inheritance (Sladek *et al.* 2007; Li and Yu 2008; Conneely and Boehnke 2007).

First, the PS may lead to many false-positive findings because of the ancestral differences between cases and controls. The genomic control (Devlin and Roeder 1999; Zheng, Freidlin, and Gastwirth 2006), structure association (Pritchard, Stephens, Rosenberg, and Donnelly 2000; Satten, Flanders, and Yang 2001), and component-based analysis (Price *et al.* 2006; Li and Yu 2008; Hibar *et al.* 2015) are three main types of procedures to correct for the PS. The genomic control could be inadequate or superfluous for adjusting for the PS, whereas the structure association is time consuming. The component-based analyses including the principal component analysis (PCA, Price *et al.* 2006) and the multidimensional scaling (Li and Yu 2008) are computationally feasible to handle considerably numerous markers and have been widely used in current genome-wide association studies. In the PCA, Price *et al.* (2006) proposed the use of several eigenvectors to represent the ancestral differences between cases and controls. Li and Yu (2008) proved that these eigenvectors are the common principal components if the similarity measure suggested in Price *et al.* (2006) is adopted. However, if other similarity measures, such as the Hamming distance, are adopted, the eigenvectors might not be the principal components. Therefore, Li and Yu (2008) proposed the principal coordinates of clusterings (PCOC) procedure, which uses the techniques from the multidimensional scaling (Mardia, Kent, and Bibby 2003) and clustering (Kaufmann and Rousseeuw 1990) methods to correct for the PS. The PCOC could be considered as an extension of the PCA.

Second, the Tracy-Widom (TW) test was proposed by Tracy and Widom (1994) to evaluate the significant eigenvalues of a matrix. It could be used to select the important principal components in the PCA. In conventional PCA approaches, the contribution rate is often adopted. However, this rate follows the rule of thumb and cannot provide the statistical significance. The TW test could remedy this defect.

Third, the distance regression (DR), which was proposed by McArdle and Anderson (2001) to analyze the multispecies responses in multifactorial ecological experiments, could be adopted to do the multiple-marker analysis (Lin and Schaid 2009; Wessel and Schork 2006) and to test the association between gene expression patterns and related variables (Zapala and Schork 2006). The original DR prohibits adjustments for the covariates. Li *et al.* (2009) extended the original DR to support adjustments for the covariates. In addition, they proposed an efficient Monte Carlo algorithm to evaluate the statistical significance and used the extended DR to select the important principal components or principal coordinates.

Fourth, the single-marker analysis, which tests for one SNP each time, is commonly used in genome-wide association studies, multiple-SNP analyses, and gene- or pathway-based procedures. The multiple-SNP, the gene-based, and the pathway-based analyses all test the association between a phenotype and many SNPs simultaneously. Some authors have developed p value combination methods, where the p values are calculated based on the single-marker analysis (Li and Yu 2008; Hu, Zhang, Zhang, Ma, and Li 2016; Zaykin *et al.* 2002). The Cochran-Armitage trend test (Sasieni 1997) and the Wald test derived for the additive model are often used in the single-marker analysis, but they may not be robust under other modes of inheritance, such as the recessive and dominant models. In addition, the Wald test is an asymptotic test. Under certain regularity conditions (Shao 2007), the Wald test statistic converges in distribution to a Chi-square distribution under the null hypothesis. The MAX3, a robust test based on the maximum of three trend tests derived from the recessive, additive,

and dominant models, has been used to identify the genetic variants associated with type II diabetes (Sladek *et al.* 2007). The MAX3 test has been included in the SAS **JMP** Genomics Software (SAS Institute Inc. 2008). However, it was based on the results of Freidlin, Zheng, Li, and Gastwirth (2002) and did not support adjustments for the covariates. Li, Zheng, Li, and Yu (2008) employed the generalized equation to obtain the MAX3 test, which could support adjustments for the covariates.

Finally, the linear regression model is a classical approach to evaluate the association between genetic variants and a quantitative trait when the quantitative trait variable follows a normal distribution. However, if the normal assumption for the quantitative trait variable does not hold, non-parametric tests such as the Kruskal-Wallis test (Kruskal and Wallis 1952) and the Jonckheere-Terpstra test (Jonckheere 1954; Terpstra 1952), are preferred. Recently, Zhang and Li (2015) proposed a non-parametric trend test (NPT) considering the genetic mode of inheritance and showed that it is more powerful than the Kruskal-Wallis test and the Jonckheere-Terpstra test. They also provided a robust non-parametric maximum test (NMAX3), which is free from the genetic models.

In this article, we introduced a new R (R Core Team 2020) package, **AssocTests** (Wang, Zhang, Li, and Zhu 2020), which provides some procedures focusing on genetic association studies. The package implements the following methods: population stratification correction through eigenvectors (EIGENSTRAT; Price *et al.* 2006), PCOC, TW test, DR, single-marker test, MAX3, NPT, and NMAX3. The trait values for these methods should be discrete or continuous. The discrete traits can be coded by 1/0 for cases/controls. The genotype values can be 0, 1, or 2 indicating the number of risk alleles for a biallelic SNP. Many packages for genetic association studies are reported. Some packages such as **GenABEL** (Aulchenko 2013), **pbatR** (Hoffmann 2018), and **snpMatrix** (Clayton and Leung 2008), provide functions to perform genome-wide association studies. The function `egscore()` in the package **GenABEL** could perform EIGENSTRAT, which is also involved in our package **AssocTests** (Wang *et al.* 2020). Some packages such as **gap** (Zhao 2007), **tdthap** (Clayton 2013), and **Rassoc** (Zang, Fung, and Zheng 2010), provide functions to test the association between individual genetic markers and a phenotype. **gap** supports the genetic data analysis of both population and family data, **tdthap** is designed for the transmission/disequilibrium tests for extended marker haplotypes, and **Rassoc** provides functions to perform robust tests for case/control genetic association studies. However, the procedures PCOC, TW test, DR with adjustments for the covariates, MAX3 test with adjustments for the covariates, NPT, and NMAX3 in our package **AssocTests** are not included in any of these packages. Package **AssocTests** is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=AssocTests`.

This paper is organized as follows. Section 2 summarizes the methods from which the **AssocTests** package was developed and also describes the functions in the package. Section 3 illustrates the capabilities of **AssocTests** by using some simulation data sets. Section 4 provides a real data example related to the PS. Finally, Section 5 concludes the paper.

2. The R package AssocTests

The procedures provided in the R package **AssocTests** are `eigenstrat()`, `pcoc()`, `tw()`, `dr()`, `smt()`, `max3()`, `npt()`, and `nmax3()`. The `eigenstrat()` procedure was developed to correct

for the PS in genetic association studies by searching for some "top" eigenvectors (Price *et al.* 2006). The `pcoc()` procedure could correct for the PS by identifying the clustering and continuous patterns of the genetic variation (Li and Yu 2008). The `tw()` procedure is based on the TW test and could evaluate the significant eigenvalues of a matrix (Tracy and Widom 1994). The `dr()` procedure is based on the DR and could detect the association between gene patterns and some independent variants of interest with or without adjustments for the covariates (Li *et al.* 2009). The `smt()` procedure is an implementation of the single-marker analysis used to identify the association between a genotype and a trait (Hoh and Ott 2003; Marchini, Donnelly, and Cardon 2005). The `max3()` procedure is a robust test to identify the association between a SNP and a binary phenotype with or without adjustments for the covariates (Sladek *et al.* 2007; Li *et al.* 2008). Finally, the `npt()` and `nmax3()` procedures perform the NPT and the robust NMAX3, which are against the normal assumption and the genetic uncertainty (Zhang and Li 2015), respectively.

2.1. Function index

The function index of the package **AssocTests** is listed as follows:

- `eigenstrat()`: EIGENSTRAT for correcting for the PS.
- `pcoc()`: Principal coordinates of clusterings for correcting for the PS.
- `tw()`: Tracy-Widom test.
- `dr()`: Distance regression.
- `smt()`: Single-marker test.
- `max3()`: Maximum test based on the maximum value of the three Cochran-Armitage trend tests under the recessive, additive, and dominant models.
- `npt()`: Non-parametric trend test based on the non-parametric risk under a given genetic model.
- `nmax3()`: NMAX3 test based on the maximum value of the three NPTs under the recessive, additive, and dominant models.

More details about them are described below.

2.2. Function `eigenstrat()`

The EIGENSTRAT for detecting and correcting for the PS provides the test through searching for the eigenvectors of the similarity matrix among the subjects in population-based genetic association studies (Price *et al.* 2006). The function `eigenstrat()` calculates the top eigenvectors or the eigenvectors with significant eigenvalues of the similarity matrix among the subjects to infer the potential population structure. It is used as follows:

```
eigenstrat(genoFile, outFile.Robj = "out.list", outFile.txt = "out.txt",
  rm.marker.index = NULL, rm.subject.index = NULL, miss.val = 9,
  num.splits = 10, topK = NULL, signt.eigen.level = 0.01,
  signal.outlier = FALSE, iter.outlier = 5, sigma.thresh = 6)
```

The arguments of the function are described as follows:

- `genoFile`: A text file containing the genotypes (0, 1, 2, or 9). The element of the file in row i and column j represents the genotype at the i-th marker of the j-th subject. 0, 1, and 2 denote the number of risk alleles, and 9 (default) is for the missing genotype.
- `outFile.Robj`: The name of an R object for saving the list of the results. Such list is the same as the return value of this function. The default is `out.list`.
- `outFile.txt`: A text file for saving the eigenvectors corresponding to the top significant eigenvalues.
- `rm.marker.index`: A numeric vector containing the indices of the removed markers. The default is `NULL`.
- `rm.subject.index`: A numeric vector containing the indices of the removed subjects. The default is `NULL`.
- `miss.val`: The value used to fill in the blanks caused by missing values in the input data. The default is `9`. The element `9` representing the missing data in the `genoFile` should be replaced according to the value of `miss.val`.
- `num.splits`: The number of groups into which the markers are split. The default is `10`.
- `topK`: The number of eigenvectors to return. If it is `NULL`, it is calculated by the TW test. The default is `NULL`.
- `signt.eigen.level`: A numeric value indicating the significance level of the TW test. It should be `0.05`, `0.01`, `0.005`, or `0.001`. The default is `0.01`.
- `signal.outlier`: A logical value indicating whether the function searches for and deletes outliers of the subjects. The default is `FALSE`.
- `iter.outlier`: A numeric value indicating the maximum iteration number for the outlier detection of the subjects. The default is `5`.
- `sigma.thresh`: A numeric value indicating the threshold for the outlier elimination. The default is `6`.

The arguments `rm.marker.index` and `rm.subject.index` could be provided according to the user-specified rules for data cleaning instances, such as an individual with excessively many missing genotype values. The argument `num.splits` does not affect the results of the EIGENSTRAT. It is used to reduce the working set (i.e., the amount of memory that an application requires) when we scan the file given by `genoFile`. Large `num.splits` results in the need for a small working set and results in slow `eigenstrat()` function run. The same usage is observed for `num.splits` in the function `pcoc()`.

This function returns a list of `num.markers` (the number of markers excluding the removed markers), `num.subjects` (the number of subjects excluding the outliers), `rm.marker.index` (the indices of the removed markers), `rm.subject.index` (the indices of the removed subjects), `TW.level` (the significance level of the TW test), `signal.outlier` (indicating whether the function deletes the outliers of the subjects), `iter.outlier` (the maximum iteration

number for the outlier detection), `sigma.thresh` (the threshold for the outlier elimination), `num.outliers` (the number of the outliers), `outliers.index` (the indices of the outliers), `num.used.subjects` (the number of the used subjects), `used.subjects.index` (the indices of the used subjects), `similarity.matrix` (the similarity matrix among the subjects), `eigenvalues` (the eigenvalues of the similarity matrix), `eigenvectors` (the eigenvectors corresponding to the eigenvalues), `topK` (the number of the significant eigenvalues), `TW.stat` (the observed values of the TW statistics), `topK.eigenvalues` (the top eigenvalues), `topK.eigenvectors` (the eigenvectors corresponding to the top eigenvalues), and `runtime` (the execution time of this function).

2.3. Function `pcoc()`

The PCOC for correcting for the PS identifies the clustering and continuous patterns of the genetic variation. The function `pcoc()` calculates the principal coordinates and the clustering of the subjects in the PCOC for correcting for the PS. It is used as follows:

```
pcoc(genoFile, outFile.txt = "pcoc.result.txt", n.MonteCarlo = 1000,
  num.splits = 10, miss.val = 9)
```

Most of the arguments are the same as those in `eigenstrat()`, and the different ones are as follows:

- `outFile.txt`: A text file for saving the results of this function. The default value is `"pcoc.result.txt"`.
- `n.MonteCarlo`: The repeat number of the Monte Carlo sampling procedure. The default is `1000`.

This function returns a list with elements `principal.coordinates` and `cluster`, where `principal.coordinates` stores the principal coordinates and `cluster` stores the clustering of the subjects. If the number of the clusters is only one, `cluster` is omitted.

2.4. Function `tw()`

The TW test detects the significant eigenvalues of a matrix. The function `tw()` calculates the number of significant eigenvalues and the TW statistics. This function was written by Bejan (2005, 2008) and is used as follows:

```
tw(eigenvalues, eigenL, criticalpoint = 2.0234)
```

The arguments of the function are described as follows:

- `eigenvalues`: A numeric vector whose elements are the eigenvalues of a matrix. The values should be sorted in a descending order.
- `eigenL`: The number of the eigenvalues.
- `criticalpoint`: A numeric value corresponding to the significance level. It should be set to `0.9793`, `2.0234`, `2.4224`, or `3.2724`, corresponding to the significance levels of 0.05, 0.01, 0.005, or 0.001, respectively (Bejan 2008). The default is `2.0234`.

This function returns a list with the class 'htest' containing statistic (a vector of the TW statistics), alternative (a character string describing the alternative hypothesis), method (a character string indicating the type of the test performed), data.name (a character string providing the name of the data), and SigntEigenL (the number of significant eigenvalues).

2.5. Function dr()

The pseudo F statistic based on the DR with or without adjustments for the covariates detects the association between a distance matrix and some independent variants of interest. A distance matrix can be transformed into a similarity matrix easily. The function dr() calculates the observed value of the test statistic and the p value of the test by using the pseudo F statistic based on the DR. It is used as follows:

```
dr(simi.mat, null.space, x.mat, permute = TRUE, n.MonteCarlo = 1000,
  seed = NULL)
```

The arguments of the function are described as follows:

- simi.mat: The similarity matrix among the subjects.
- null.space: A numeric vector containing the indices of those columns in x.mat corresponding to the null space.
- x.mat: The covariate matrix which combines the null space and the matrix of interest.
- permute: A logical value indicating whether the Monte Carlo sampling procedure is invoked without replacement. The default is TRUE.
- n.MonteCarlo: The repeat number of the Monte Carlo sampling procedure. The default is 1000.
- seed: The seed of the random number generator. The default is NULL.

This function returns a list with the class 'htest' containing statistic (the observed value of the test statistic), p.value (the p value of the test), alternative (a character string describing the alternative hypothesis), method (a character string indicating the type of the test performed), and data.name (a character string describing the names of the data). The return values of the functions max3(), npt(), and nmax3() described below are similar to that of this function.

2.6. Function smt()

The single-marker test is used to identify the association between the genotype at a biallelic marker and a trait using the Wald test or the Fisher's exact test. The function smt() calculates the number of the valid subjects and the p value of the single-marker test. It is used as follows:

```
smt(y, g, covariates = NULL, min.count = 5, missing.rate = 0.20,
  y.continuous = FALSE)
```

The arguments of the function are described as follows:

- `y`: A numeric vector of the observed trait values in which the i-th element is for the i-th subject. The elements could be discrete (`0` or `1`) or continuous. Any missing value is represented by `NA`.
- `g`: A numeric vector of the observed genotype values (`0`, `1`, or `2`, denoting the number of risk alleles) in which the i-th element is for the i-th subject. Any missing value is represented by `NA`. `g` has the same length as `y`.
- `covariates`: An optional data frame, list, or environment containing the covariates used in the model. The default is `NULL`, that is, no covariates are present.
- `min.count`: A threshold to decide which method is used to calculate the p value when the trait is discrete and `covariates = NULL`. For a certain genotype and a certain trait value, we have a corresponding number of the subjects. If the minimum value of all such numbers traversing all possible genotypes and trait values is less than `min.count`, the Fisher's exact test is adopted; otherwise, the Wald test is adopted. The default is `5`.
- `missing.rate`: The highest missing value rate of the genotype values that this function can tolerate. The default is `0.20`.
- `y.continuous`: A logical value indicating whether `y` is continuous. The default is `FALSE`.

If `y` is continuous, this function returns a list with the class '`htest`', which contains the components `statistic`, `p.value`, `alternative`, `method`, `data.name`, and `sample.size`. The components `statistic`, `p.value`, `alternative`, `method`, and `data.name` are similar to those of `dr()`. `sample.size` is a vector providing the numbers of the subjects with the genotypes `0`, `1`, and `2` (`n0`, `n1`, and `n2`, respectively). If `y` is discrete, this function returns a list with the class '`htest`' containing the components `statistic`, `p.value`, `alternative`, `method`, `data.name`, `sample.size`, and `bad.obs`. The components `statistic`, `p.value`, `alternative`, `method`, and `data.name` are similar to those of `dr()`. Meanwhile, `sample.size` is a vector that provides the number of the subjects with the trait value `1` and the genotype `0` (`r0`), the number of the subjects with the trait value `1` and the genotype `1` (`r1`), the number of the subjects with the trait value `1` and the genotype `2` (`r2`), the number of the subjects with the trait value `0` and the genotype `0` (`s0`), the number of the subjects with the trait value `0` and the genotype `1` (`s1`), and the number of the subjects with the trait value `0` and the genotype `2` (`s2`). `bad.obs` is a vector that provides the number of the missing genotype values with the trait value `1` (`r.miss`), the number of the missing genotype values with the trait value `0` (`s.miss`), and the total number of missing genotype values (`n.miss`).

2.7. Function `max3()`

The MAX3 test based on the trend tests without adjustments for the covariates or based on the Wald tests with adjustments for the covariates is conducted for the association between a SNP and a binary phenotype. The test statistic is the maximum value of the three test statistics derived for the recessive, additive, and dominant models. The function `max3()` calculates the observed value of the MAX3 statistic and the p value of the MAX3 test. It is used as follows:

```
max3(y, g, covariates = NULL, Score.test = TRUE, Wald.test = FALSE,
  rhombus.formula = FALSE)
```

The arguments of the function are described as follows:

- `y`: A numeric vector of the observed trait values in which the i-th element is for the i-th subject. The elements should be either `0` or `1`.
- `g`: A numeric vector of the observed genotype values (`0`, `1`, or `2`, denoting the number of risk alleles) in which the i-th element is for the i-th subject. Any missing value is represented by `NA`. `g` has the same length as `y`.
- `covariates`: A numeric matrix specifying the covariates used in the model. Each column is for one covariate. The default is `NULL`, that is, no covariates are needed to be adjusted for.
- `Score.test`: A logical value. If it is `TRUE`, the score tests are used. Either `Score.test` or `Wald.test` should be `FALSE`, and the other should be `TRUE`. The default is `TRUE`.
- `Wald.test`: A logical value. If it is `TRUE`, the Wald tests are used. Either `Score.test` or `Wald.test` should be `FALSE`, and the other should be `TRUE`. The default is `FALSE`.
- `rhombus.formula`: A logical value. If it is `TRUE`, the p value of the MAX3 test is approximated by the rhombus formula. Otherwise the twofold integration is adopted to calculate the p value. The default is `FALSE`.

The rhombus formula is an approximation formula to estimate the two-sided test p value for the MAX3 statistic (Li *et al.* 2008). It is an extension of the W-formula, which was originally derived to estimate the one-sided test p value of the MAX3 statistic (Efron 1997).

The function `max3()` in the package **AssocTests** can test the association between a SNP and a binary phenotype with or without correcting for the covariates. This function differs from the function `MAX3()` in the package **Rassoc**, which can only test for the association without correcting for the covariates.

2.8. Function `npt()`

The NPT examines the association between a genetic variant and a non-normally distributed quantitative trait based on the non-parametric risk. The function `npt()` calculates the observed value of the NPT statistic and the p value of this test under a specific genetic model. It is used as follows:

```
npt(y, g, varphi)
```

The arguments of the function are described as follows:

- `y`: A numeric vector of the observed quantitative trait values in which the i-th element is the trait value of the i-th subject.
- `g`: A numeric vector of the observed genotype values (`0`, `1`, or `2`, denoting the number of risk alleles) in which the i-th element is the genotype value of the i-th subject for a biallelic SNP. `g` has the same length as `y`.

- `varphi`: A numeric value representing the genetic model. It should be `0`, `0.5`, or `1`, which indicates that the calculation should be performed under the recessive, additive, or dominant model, respectively.

2.9. Function `nmax3()`

When the genetic model is uncertain, a robust test is preferred. The MAX3 test is a widely-used robust test in case/control association studies. NMAX3 is a non-parametric MAX3 test based on the NPT to evaluate the association between a biallelic SNP and a quantitative trait. The function `nmax3()` calculates the observed value of the NMAX3 statistic and the p value of this test. It is used as follows:

```
nmax3(y, g)
```

The arguments `y` and `g` are the same as those in `npt()`.

The function `nmax3()` in the package **AssocTests** differs from the functions `max3()` described above in the package **AssocTests** and `MAX3()` in the package **Rassoc**. `nmax3()` is constructed on the basis of the NPT for quantitative trait association studies, whereas `max3()` and `MAX3()` are used for case/control association studies and derived from the Cochran-Armitage trend test.

3. Simulation examples

Some simulation examples are used to illustrate the usages and capabilities of the functions in **AssocTests** (Wang *et al.* 2020). The analyses were conducted using the R version 3.6.3 (R Core Team 2020).

The data sets used in this section and the next section have been placed into a data-only package **AssocTests.data** (Wang, Zhang, Li, and Zhu 2015). This package can be downloaded from `https://github.com/statscueb/AssocTests.data` or use the function `install_github()` in the package **devtools** (Wickham, Hester, and Chang 2019) to install it directly.

```
R> library("devtools")
R> install_github("statscueb/AssocTests.data")
R> library("AssocTests.data")
R> help(package = "AssocTests.data")
```

The data sets contained in package **AssocTests.data** are `arthritisG`, `arthritisP`, `drG`, `drP`, `drS`, `extreme2PSG`, `extreme2PSP`, `extreme3PSG`, `extreme3PSP`, `moderate2PSG`, `moderate2PSP`, `moderate3PSG`, and `moderate3PSP`.

3.1. Simulation: `eigenstrat()` and `tw()`

The simulation data set consists of 1,000 cases and 1,000 controls. For each individual, we generate the genotypes of 10,000 SNPs that are not associated with the disease. Following Price *et al.* (2006) and Li and Yu (2008), we consider two population substructures for the study population as follows: S1 (two underlying discrete subpopulations) and S2 (three underlying

Substructure	Level	Case proportion	Control proportion
S1	moderate	(0.6, 0.4)	(0.4, 0.6)
S1	extreme	(0.5, 0.5)	(0, 1)
S2	moderate	(0.45, 0.35, 0.20)	(0.35, 0.20, 0.45)
S2	extreme	(0.33, 0.67, 0)	(0, 0.33, 0.67)

Table 1: The PS for discrete population substructures S1 and S2.

discrete subpopulations). The Hardy-Weinberg equilibrium (HWE) within each subpopulation is assumed. The allele frequency for each SNP is generated from the Beta distribution with the parameters $p(1 - F_{ST})/F_{ST}$ and $(1 - p)(1 - F_{ST})/F_{ST}$ in which the inbreeding coefficient F_{ST} is 0.01 and the ancestry population allele frequency p is drawn from the uniform distribution on $[0.1, 0.9]$. In each population substructure, we consider two levels of the ancestral differences between the cases and the controls, which are moderate and extreme, by varying the sampling fractions summarized in Table 1. For the first simulation example, the population substructure is S1, and the level of the ancestral differences between the cases and the controls is moderate. In the package **AssocTests.data**, the data sets `moderate2PSG` and `moderate2PSP` are the genotype data and the phenotype data, respectively, under this condition. We save the data set `moderate2PSG` in a text file which can be used as the input of the function `eigenstrat()`.

```
R> data("moderate2PSG", package = "AssocTests.data")
R> data("moderate2PSP", package = "AssocTests.data")
R> gFile <- "moderate2PSG.txt"
R> write.table(moderate2PSG, file = gFile, quote = FALSE, sep = "",
+    row.names = FALSE, col.names = FALSE)
```

In the function `eigenstrat()`, accordingly, we know that `genoFile = gFile`. We consider that `outFile.Robj = "moderate2PS.E.list"` and `outFile.txt = "moderate2PS.E.txt"`. `signt.eigen.level` is set to `0.05` and the other arguments are set to their default values.

```
R> result.E <- eigenstrat(genoFile = gFile,
+    outFile.Robj = "moderate2PS.E.list",
+    outFile.txt = "moderate2PS.E.txt", signt.eigen.level = 0.05)
R> result.E$topK

[1] 1

R> n <- length(result.E$eigenvalues)
R> n

[1] 2000
```

In function `tw()`, we use `result.E$eigenvalues[1:(n - 1)]` as the value of `eigenvalues` and `n - 1` as the value of `eigenL`. The `criticalpoint` is set to `0.9793`, which corresponds to the significance level of 0.05.

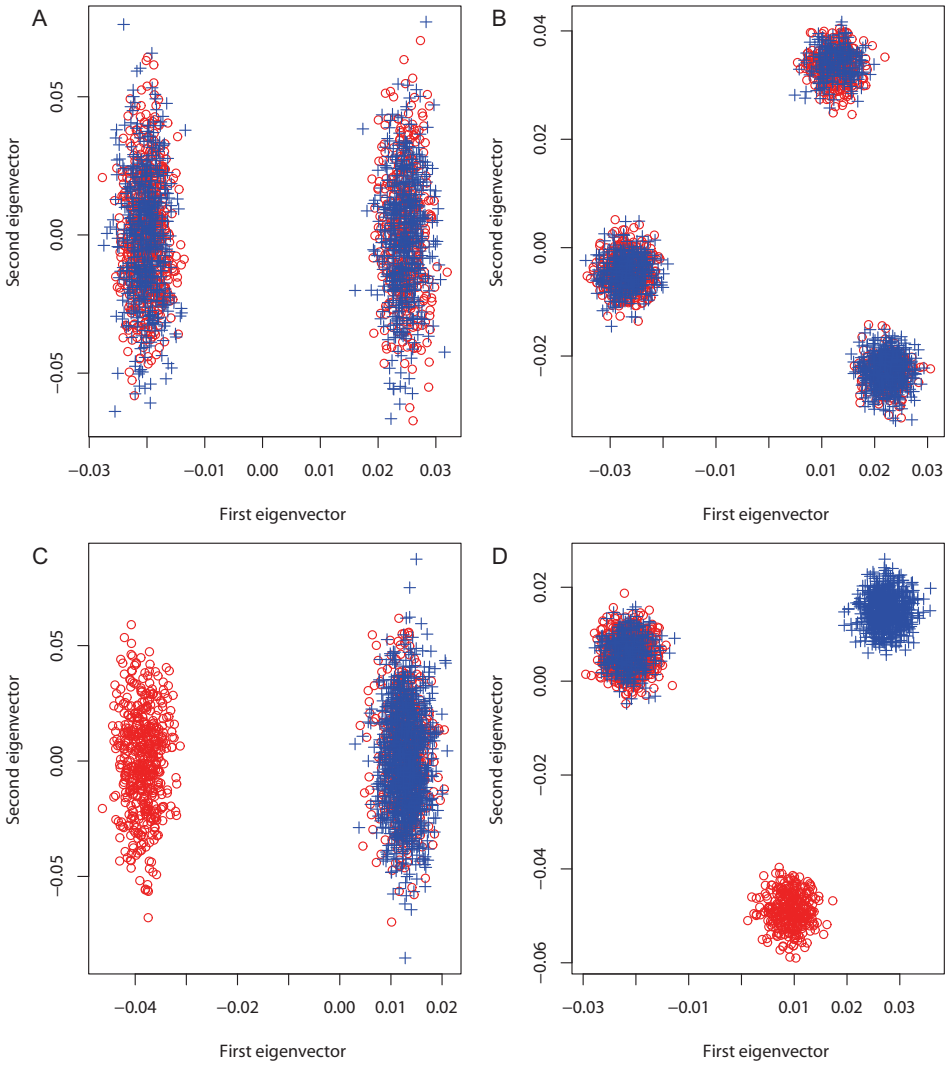

Figure 1: Samples in the space of the first two eigenvectors. (A)–(D) are for the samples corresponding to the first, second, third, and fourth examples, respectively. The red circles and the blue pluses represent the cases and the controls, respectively.

```
R> cp <- 0.9793
R> result.TW <- tw(eigenvalues = result.E$eigenvalues[1:(n - 1)],
+    eigenL = n - 1, criticalpoint = cp)
R> result.TW$SigntEigenL

[1] 1
```

The value of `result.E$topK` from the function `eigenstrat()` is 1, which is consistent with the value of `result.TW$SigntEigenL` from the function `tw()`. Thus, the number of significant eigenvalues is 1 in this example. For the second simulation example, the population substructure is S2, and the level of the ancestral differences between the cases and the controls is moderate. In the package **AssocTests.data** (Wang *et al.* 2015), the data sets `moderate3PSG`

and moderate3PSP are the genotype data and the phenotype data, respectively, generated under this situation. For the third simulation example, the population substructure is S1, and the level of the ancestral differences between the cases and the controls is extreme. In the package **AssocTests.data**, the data sets extreme2PSG and extreme2PSP are the generated genotype and phenotype data, respectively. For the fourth simulation example, the population substructure is S2, and the level of the ancestral differences between the cases and the controls is extreme. In the package **AssocTests.data**, the data sets extreme3PSG and extreme3PSP are the generated genotype and phenotype data, respectively. The R codes for the second, third, and fourth simulation examples are similar to those for the first one.

The results of result.E$topK from the function eigenstrat() are 1, 2, 1, and 2 for the four examples, respectively, conforming to the results of result.TW$SigntEigenL from the function tw(). Hence, the numbers of the significant eigenvalues are 1, 2, 1, and 2, respectively. Figure 1 plots the samples in the space of the first two eigenvectors of the similarity matrices for the four simulation examples. The clustering patterns are noticeable, with the subjects from the same subpopulation staying close. Furthermore, the distributions of cases (represented by red circles) are nonuniform in controls (represented by blue pluses), especially in the third and fourth examples (Figure 1 C and D), illustrating the allele frequency differences between the cases and the controls due to systematic ancestry differences, that is, the PS.

3.2. Simulation: pcoc()

The simulation design and the examples for the function pcoc() are the same as those for eigenstrat(). For the first example, genoFile = gFile. Furthermore, outFile is set to "moderate2PS.PCOC.txt". The other arguments are all set to their default values.

```
R> result.PCOC <- pcoc(genoFile = gFile, outFile = "moderate2PS.PCOC.txt")
```

The R codes for the second, third, and fourth simulation examples are similar to those for the first one.

We can calculate the accuracies of the clusterings provided by the function pcoc() by using the values of result.PCOC$cluster, considering that we know the true clusterings of the subjects in the simulation design. We find that pcoc() can classify the subpopulations with 100% accuracy in the four examples, where the subpopulation patterns are tremendously recognizable and no overlap exists between different subpopulations.

3.3. Simulation: dr()

Considering the linkage disequilibriums among the SNPs, we use the real data set that contains the genotypes of 127 SNPs in the uronyl-2-sulfotransferase gene from Genetic Analysis Workshop 16 (Cupples *et al.* 2009; Amos *et al.* 2009; Lin *et al.* 2009) to generate the simulation data set. After the deletion of the subjects containing missing values, the real data set consists of 1,081 subjects. The disease prevalence is set to 0.05 and the first 50 SNPs are assumed to be associated with the disease with a log odds ratio ln(1.05). We use the model $\ln \frac{p_j}{1-p_j} = \ln(0.05/0.95) + \ln(1.05) \times x_{j1} + \cdots + \ln(1.05) \times x_{j50}$ $(j = 1, \cdots, 1081)$ to simulate the phenotypes of the subjects. In the package **AssocTests.data** (Wang *et al.* 2015), the data sets drG, drP, and drS are the genotype data, phenotype data, and the similarity matrix of the genotype data, respectively. In the function dr(), the left and the right parts

of the argument `x.mat` are set to a 1,081-dimensional column vector of 1s and the vector of the phenotype values, respectively. The `null.space` stores the column indices of the left part of `x.mat`. The other arguments are all set to their default values.

```
R> data("drP", package = "AssocTests.data")
R> data("drS", package = "AssocTests.data")
R> set.seed(100)
R> x.mat <- cbind(rep(1, length(drP)), drP)
R> result.DR <- dr(simi.mat = drS, null.space = 1, x.mat)
R> result.DR

          Distance regression

data:  drS and x.mat
F = 0.0018221, p-value = 0.011
alternative hypothesis: the pair-wise similarity is influenced by the
variants of interest
```

This test is two-sided. The null hypothesis is that all the regression coefficients are 0s, that is, the pair-wise similarity is not influenced by the variants of interest. The alternative hypothesis is that some regression coefficients are nonzero, that is, the pair-wise similarity is influenced by the variants of interest. The result indicates that the p value is less than 0.05, illustrating that the markers are associated with the disease, conforming to the simulation design.

3.4. Simulation: `smt()`

The simulation design in this section is similar to that in Li *et al.* (2008). The simulation data set consists of 1,000 cases and 1,000 controls. HWE is assumed and the minor allele frequency (MAF) is set to 0.3. Furthermore, the additive model is considered. The relative risks of the groups with genotypes 1 and 2 relative to the group with genotype 0 are 1.2 and 1.4, respectively. The disease prevalence is set to 0.05.

```
R> ncases <- 1000
R> ncontrols <- 1000
R> y <- rep(c(1, 0), c(ncases, ncontrols))
R> g <- rep(2, ncases + ncontrols)
R> MAF <- 0.3
R> rr10 <- 1.2
R> rr20 <- 1.4
R> dp <- 0.05
R> x <- dp / ((1 - MAF)^2 + rr10 * 2 * MAF * (1 - MAF) + rr20 * MAF^2)
R> a <- round(x * (1 - MAF)^2 / dp * ncases)
R> b <- round(rr10 * x * 2 * MAF * (1 - MAF) / dp * ncases)
R> d <- round((1 - x) * (1 - MAF)^2 /(1 - dp) * ncontrols)
R> e <- round((1 - rr10 * x) * 2 * MAF * (1 - MAF) / (1 - dp) * ncontrols)
R> g[1:a] <- 0
R> g[(a + 1):(a + b)] <- 1
```

```
R> g[(ncases + 1):(ncases + d)] <- 0
R> g[(ncases + d + 1):(ncases + d + e)] <- 1
```

We can use the function `smt()` to test the association between `y` and `g`.

```
R> result.SMT <- smt(y, g)
R> result.SMT

          Single-marker test

data:  y and g
p-value = 0.006666
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

The p value of the test illustrates that the genotype and the phenotype in this example are significantly associated, with the significance level of 0.05.

3.5. Simulation: `max3()`

The simulation data sets of the first example for `max3()` are the same as those for `smt()`. We can use the function `max3()` to test the association between the genotype `g` and the phenotype `y` from Section 3.4.

```
R> max3(y, g, covariates = NULL, Score.test = FALSE, Wald.test = TRUE,
+    rhombus.formula = FALSE)

         MAX3 test

data:  y and g
MAX3 = 2.7169, p-value = 0.0152
alternative hypothesis: the phenotype is significantly associated with the
genotype

R> max3(y, g, covariates = NULL, Score.test = FALSE, Wald.test = TRUE,
+    rhombus.formula = TRUE)

          MAX3 test

data:  y and g
MAX3 = 2.7169, p-value = 0.01515
alternative hypothesis: the phenotype is significantly associated with the
genotype

R> max3(y, g, covariates = NULL, Score.test = TRUE, Wald.test = FALSE,
+    rhombus.formula = FALSE)
```

```
        MAX3 test

data:  y and g
MAX3 = 2.7169, p-value = 0.0152
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

```
R> max3(y, g, covariates = NULL, Score.test = TRUE, Wald.test = FALSE,
+    rhombus.formula = TRUE)

        MAX3 test

data:  y and g
MAX3 = 2.7169, p-value = 0.01515
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

The p values of the tests illustrate that significant association is found between the genotype and the phenotype in this example with the significance level of 0.05.

The simulation design of the second example for `max3()` is similar to that in Li *et al.* (2008). The simulation data sets consist of two subpopulations, the sample sizes of which are both 1,000. The proportions of the cases from the two subpopulations are 0.6 and 0.4, respectively, whereas those of the controls from the two subpopulations are 0.4 and 0.6, respectively. Therefore, the numbers of the cases and the controls are both 1,000. HWE is assumed within each subpopulation. The MAFs are 0.3 and 0.35 for the two subpopulations, respectively. The additive model is considered. The relative risks of the groups with genotypes 1 and 2 relative to the group with genotype 0 are 1.2 and 1.4, respectively, within each subpopulation. The disease prevalence is set to 0.05 for the two subpopulations.

```
R> n.sp1 <- 1000
R> n.sp2 <- 1000
R> ncases.sp1 <- n.sp1 * 0.6
R> ncases.sp2 <- n.sp2 * 0.4
R> ncontrols.sp1 <- n.sp1 * 0.4
R> ncontrols.sp2 <- n.sp2 * 0.6
R> n <- c(ncases.sp1, ncontrols.sp1, ncases.sp2, ncontrols.sp2)
R> sn <- cumsum(n)
R> y <- rep(c(1, 0, 1, 0), n)
R> g <- rep(2, n.sp1 + n.sp2)
R> MAF <- c(0.3, 0.35)
R> rr10 <- 1.2
R> rr20 <- 1.4
R> dp <- 0.05
R> x <- dp / ((1 - MAF)^2 + rr10 * 2 * MAF * (1 - MAF) + rr20 * MAF^2)
R> a <- round(x * (1 - MAF)^2 / dp * c(ncases.sp1, ncases.sp2))
R> b <- round(rr10 * x * 2 * MAF * (1 - MAF) / dp *
+    c(ncases.sp1, ncases.sp2))
```

```
R> d <- round((1 - x) * (1 - MAF)^2 / (1 - dp) *
+    c(ncontrols.sp1, ncontrols.sp2))
R> e <- round((1 - rr10 * x) * 2 * MAF * (1 - MAF) / (1 - dp) *
+    c(ncontrols.sp1, ncontrols.sp2))
R> g[1:a[1]] <- 0
R> g[(a[1] + 1):(a[1] + b[1])] <- 1
R> g[(sn[1] + 1):(sn[1] + d[1])] <- 0
R> g[(sn[1] + d[1] + 1):(sn[1] + d[1] + e[1])] <- 1
R> g[(sn[2] + 1):(sn[2] + a[2])] <- 0
R> g[(sn[2] + a[2] + 1):(sn[2] + a[2] + b[2])] <- 1
R> g[(sn[3] + 1):(sn[3] + d[2])] <- 0
R> g[(sn[3] + d[2] + 1):(sn[3] + d[2] + e[2])] <- 1
```

We run the function `max3()` with adjustments for the subpopulation structure, namely `covariates`. `covariates` is a matrix with a size (`n.sp1` + `n.sp2`) $\times 1$, the elements of which are 0s and 1s for the subjects from the two subpopulations, respectively.

```
R> z <- matrix(rep(c(0, 1), c(n.sp1, n.sp2)), ncol = 1)
R> max3(y, g, covariates = z, Score.test = FALSE, Wald.test = TRUE,
+    rhombus.formula = FALSE)

        MAX3 test

data:  y and g
MAX3 = 2.6494, p-value = 0.01849
alternative hypothesis: the phenotype is significantly associated with the
genotype

R> max3(y, g, covariates = z, Score.test = FALSE, Wald.test = TRUE,
+    rhombus.formula = TRUE)

        MAX3 test

data:  y and g
MAX3 = 2.6494, p-value = 0.01847
alternative hypothesis: the phenotype is significantly associated with the
genotype

R> max3(y, g, covariates = z, Score.test = TRUE, Wald.test = FALSE,
+    rhombus.formula = FALSE)

        MAX3 test

data:  y and g
MAX3 = 2.6551, p-value = 0.0182
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

```
R> max3(y, g, covariates = z, Score.test = TRUE, Wald.test = FALSE,
+    rhombus.formula = TRUE)

          MAX3 test

data:  y and g
MAX3 = 2.6551, p-value = 0.01815
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

According to the results of the function `max3()`, the p values of the tests are `0.01849`, `0.01847`, `0.0182`, and `0.01815` when we choose to use the Wald test and the twofold integration, the Wald test and the rhombus formula, the score test and the twofold integration, and the score test and the rhombus formula, respectively, illustrating that a significant association is found between the marker and the phenotype with correcting for the PS in this example with the significance level of 0.05.

3.6. Simulation: `npt()` and `nmax3()`

The simulation data set consists of 1,000 subjects. Following Zhang and Li (2015), we assume that the quantitative trait y relates to a biallelic SNP with the genotype g as the linear model $y = \beta_0 + g\beta_1 + e$, where e is the error term that follows a generalized extreme value distribution, tGEV$(0, 0, 1)$, with the shape parameter 0, the location parameter 0, and the scale parameter 1. We set $\beta_0 = 0.5$, $\beta_1 = \ln 1.2$, and MAF $= 0.3$ in the population. The genetic model is assumed to be additive in this simulation.

```
R> n <- 1000
R> set.seed(100)
R> e <- rgev(n, 0, 0, 1)
R> MAF <- 0.3
R> g <- rbinom(n, 2, MAF)
R> y <- 0.5 + g * log(1.2) + e
```

We can use the function `npt()` to test the association between the quantitative trait `y` and the SNP with the genotype being `g`.

```
R> result.NPT <- npt(y, g, 0.5)
R> result.NPT

          Nonparametric trend test

data:  y and g
NPT = 4.1097, p-value = 3.962e-05
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

The p value of the NPT for the additive model is 3.962×10^{-5}, which is far less than the significance level of 0.05. Thus, the quantitative trait is significantly associated with this SNP.

We can also use the function `nmax3()` to test the association between the quantitative trait `y` and the biallelic SNP with genotype `g` by using the NMAX3.

```
R> result.NMAX3 <- nmax3(y, g)
R> result.NMAX3

        Nonparametric MAX3 test

data:  y and g
NMAX3 = 4.1097, p-value = 7.779e-05
alternative hypothesis: the phenotype is significantly associated with the
genotype
```

This result also shows that the continuous phenotype `y` is associated with the biallelic SNP by using the NMAX3 with the significance level of 0.05.

4. An application: Rheumatoid arthritis with PS

This section presents a detailed application of this package on the association analysis of rheumatoid arthritis with the PS. The data is from the Genetic Analysis Workshop 16 Problem 1 (Cupples *et al.* 2009; Amos *et al.* 2009). The genotype data set used for correcting for the PS consists of 868 cases and 1,194 controls at 12,749 SNPs that are not associated with the disease (Yu *et al.* 2008). The genotype and the phenotype data sets are saved in `arthritisG` and `arthritisP`, respectively, in the package **AssocTests.data** (Wang *et al.* 2015).

```
R> data("arthritisG", package = "AssocTests.data")
R> data("arthritisP", package = "AssocTests.data")
R> arth.gFile <- "arthritisG.txt"
R> write.table(arthritisG, file = arth.gFile, quote = FALSE, sep = "",
+    row.names = FALSE, col.names = FALSE)
```

We can use the function `eigenstrat()` to calculate the significant eigenvalues and the corresponding eigenvectors of the similarity matrix.

```
R> arth.E <- eigenstrat(genoFile = arth.gFile,
+    outFile.Robj = "arthritis.E.list", outFile.txt = "arthritis.E.txt")
R> arth.E$topK

[1] 4
```

We can also use `tw()` and `arth.E$eigenvalues[1:(nrow(arthritisP) - 1)]` to calculate the significant eigenvalues.

```
R> arth.TW <- tw(eigenvalues = arth.E$eigenvalues[1:(nrow(arthritisP) - 1)],
+    eigenL = nrow(arthritisP) - 1)
R> arth.TW$SigntEigenL

[1] 4
```

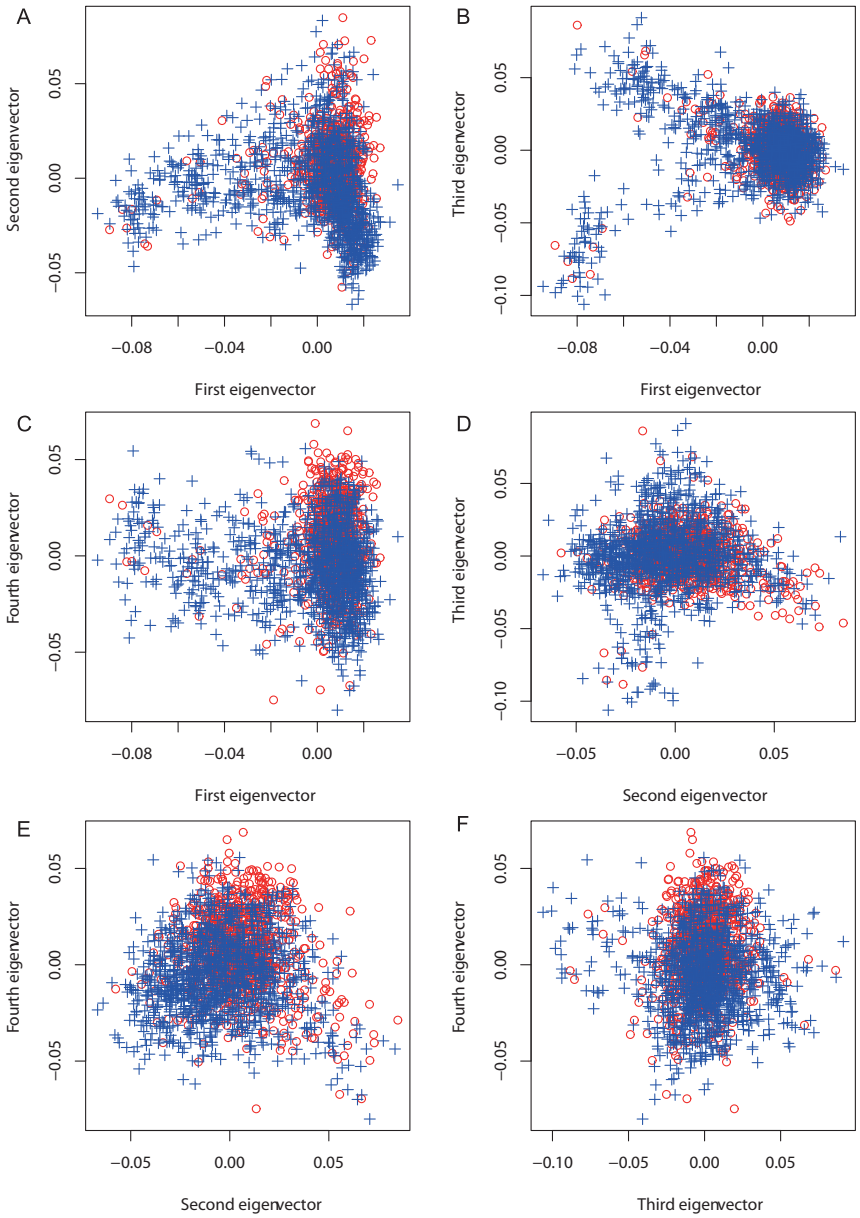

Figure 2: Samples in the space of the first four eigenvectors. The red circles and the blue pluses represent the cases and the controls, respectively.

The value of `arth.E$topK` from the function `eigenstrat()` is 4, which is consistent with the value of `arth.TW$SigntEigenL` from the function `tw()`. Thus, the number of significant eigenvalues is 4 in this example. Figure 2 plots the samples in the space of the first four eigenvectors of the similarity matrix. The distributions of the cases (represented by the red circles) are nonuniform in the controls (represented by the blue pluses), especially in Figure 2 C, E, and F, where the fourth eigenvector is involved in, illustrating that the PS exists in the data.

The function `pcoc()` can be used to detect the population structure of this example. This data set consists of two subpopulations according to the result of `pcoc()`.

```
R> arth.PCOC <- pcoc(genoFile = arth.gFile, outFile = "arthritis.PCOC.txt")
R> levels(arth.PCOC$cluster)

[1] "1" "2"
```

5. Conclusions

In this article, we have outlined some methods and algorithms for the genetic association studies and described the R package **AssocTests** (Wang *et al.* 2020), which contains the procedures EIGENSTRAT, PCOC, TW test, DR, single-marker test, MAX3 test with or without adjustments for the covariates, NPT, and NMAX3. The descriptions of the functions have their counterparts in the R package **AssocTests**.

We demonstrated the usages and the capabilities of this package in some simulation studies and real data analyses in genetic association studies. Actually, the functions can also be used in other application areas, such as food processing, economics, and finance. All the functions in this package performed well. The computational complexity is often extremely high in the genome-wide association study, typically using 500,000 $\sim$ 1,000,000 SNPs across the genome. The functions in this package are effective. Considerably numerous SNPs are feasibly handled. Although the execution time is relatively long, it is affordable. Furthermore, the multitrait genetic association study (Thoen *et al.* 2017), which is developing rapidly recently, is not involved in this package. For further works, the methods for the multitrait genetic association study will be implemented in an updated version of this package. Depending on the demands of users, we may consider developing a graphical user interface for this package.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments. The authors are very grateful to Weicheng Zhu for the valuable discussions. The authors thank Dera Tompkins, NIH Library Writing Center, for manuscript editing assistance. The work of Q. Li was supported in part by Special National Key Research and Development Plan under grant 2016YFD0400206, in part by Beijing Natural Science Foundation under grant Z180006, and in part by National Science Foundation of China under grant 11722113. The work of L. Wang was supported in part by National Nature Science Foundation of China under grant 11701391 and in part by Scientific Research Level Improvement Quota Project of Capital University of Economics and Business. Research of W. Zhang was partially supported by the 100 Talents Program of The Chinese Academy of Sciences for Young Scholars. Data of GAW16 were gathered with the support of grants NO1-AR-2-2263 and RO1-AR-44422 from the National Institutes of Health (the PI is Peter K. Gregersen) and the National Arthritis Foundation.

References

Amos CI, Chen WV, Seldin MF, Remmers EF, Taylor KE, Criswell LA, Lee AT, Plenge RM, Kastner DL, Gregersen PK (2009). "Data for Genetic Analysis Workshop 16 Problem 1, Association Analysis of Rheumatoid Arthritis Data." *BMC Proceedings*, **3**(S7), S2. doi:10.1186/1753-6561-3-s7-s2.

Ardlie KG, Deluca DS, Segrè AV, Sullivan TJ, Young TR, Gelfand ET, Trowbridge CA, Maller JB, Tukiainen T, Lek M, *et al.* (2015). "The Genotype-Tissue Expression (Gtex) Pilot Analysis: Multitissue Gene Regulation in Humans." *Science*, **348**(6235), 648–660. doi:10.3410/f.725479865.793506594.

Aulchenko Y (2013). ***GenABEL**: Genome-Wide SNP Association Analysis*. R package version 1.8-0, URL https://CRAN.R-project.org/src/contrib/Archive/GenABEL.

Bejan AI (2005). *Largest Eigenvalues and Sample Covariance Matrices*. Master's thesis, University of Warwick, United Kingdom.

Bejan AI (2008). "Tracy-Widom and Painleve II: Computational Aspects and Realisation in S-PLUS." In *First Workshop of the ERCIM Working Group on Computing and Statistics*. Neuchatel, Switzerland.

Burton PR, Clayton DG, Cardon LR, Craddock N, Deloukas P, Duncanson A, Kwiatkowski DP, McCarthy MI, Ouwehand WH, Samani NJ, *et al.* (2007). "Genome-Wide Association Study of 14,000 Cases of Seven Common Diseases and 3,000 Shared Controls." *Nature*, **447**(7145), 661–678. doi:10.1038/nature05911.

Clayton D (2013). ***tdthap**: TDT Tests for Extended Haplotypes*. R package version 1.1-7, URL https://CRAN.R-project.org/package=tdthap.

Clayton D, Leung HT (2008). ***snpMatrix**: The 'snp.matrix' and 'X.snp.matrix' Classes*. R package version 1.6-1, URL http://www-gene.cimr.cam.ac.uk/clayton/software/.

Conneely KN, Boehnke M (2007). "So Many Correlated Tests, So Little Time! Rapid Adjustment of *P* Values for Multiple Correlated Tests." *The American Journal of Human Genetics*, **81**(6), 1158–1168. doi:10.1086/522036.

Cupples LA, Beyene J, Bickeboller H, Daw EW, Fallin MD, Gauderman WJ, Ghosh S, Goode EL, Hauser ER, Hinrichs A, KentJr JW, Martin LJ, Martinez M, Neuman RJ, Province M, Szymczak S, Wilcox MA, Ziegler A, MacCluer JW, Almasy L (2009). "Genetic Analysis Workshop 16: Strategies for Genome-Wide Association Study Analyses." *BMC Proceedings*, **3**(S7), S1. doi:10.1186/1753-6561-3-s7-s1.

Devlin B, Roeder K (1999). "Genomic Control for Association Studies." *Biometrics*, **55**(4), 997–1004. doi:10.1111/j.0006-341x.1999.00997.x.

Dudbridge F, Koeleman BPC (2004). "Efficient Computation of Significance Levels for Multiple Associations in Large Studies of Correlated Data, Including Genomewide Association Studies." *The American Journal of Human Genetics*, **75**(3), 424–435. doi:10.1086/423738.

Efron B (1997). "The Length Heuristic for Simultaneous Hypothesis Tests." *Biometrika*, **84**(1), 143–157. doi:10.1093/biomet/84.1.143.

Freidlin B, Zheng G, Li Z, Gastwirth JL (2002). "Trend Tests for Case-Control Studies of Genetic Markers: Power, Sample Size and Robustness." *Human Heredity*, **53**, 146–152. doi:10.1159/000064976.

Hibar DP, Stein JL, Renteria ME, Arias-Vasquez A, Desrivières S, Jahanshad N, Toro R, Wittfeld K, Abramovic L, Andersson M, *et al.* (2015). "Common Genetic Variants Influence Human Subcortical Brain Structures." *Nature*, **520**(7546), 224–229. doi:10.1038/nature14101.

Hoffmann T (2018). ***pbatR**: Pedigree/Family-Based Genetic Association Tests Analysis and Power*. R package version 2.2-13, URL https://CRAN.R-project.org/package=pbatR.

Hoh J, Ott J (2003). "Mathematical Multi-Locus Approaches to Localizing Complex Human Trait Genes." *Nature Reviews Genetics*, **4**(9), 701–709. doi:10.1038/nrg1155.

Hoh J, Wille A, Ott J (2001). "Trimming, Weighting, and Grouping SNPs in Human Case-Control Association Studies." *Genome Research*, **11**(12), 2115–2119. doi:10.1101/gr.204001.

Hu X, Zhang W, Zhang S, Ma S, Li Q (2016). "Group-Combined P Values with Applications to Genetic Association Studies." *Bioinformatics*, **32**(18), 2737–2743. doi:10.1093/bioinformatics/btw314.

Jonckheere AR (1954). "A Distribution-Free K-Sample Test against Ordered Alternatives." *Biometrika*, **41**(1–2), 133–145. doi:10.1093/biomet/41.1-2.133.

Kaufmann L, Rousseeuw PJ (1990). *Finding Groups in Data*. John Wiley & Sons. doi:10.1002/9780470316801.

Klein RJ, Zeiss C, Chew EY, Tsai JY, Sackler RS, Haynes C, Henning AK, SanGiovanni JP, Mane SM, Mayne ST, *et al.* (2005). "Complement Factor H Polymorphism in Age-Related Macular Degeneration." *Science*, **308**(5720), 385–389. doi:10.1126/science.1109557.

Kruskal WH, Wallis WA (1952). "Use of Ranks in One-Criterion Variance Analysis." *Journal of the American Statistical Association*, **47**(260), 583–621. doi:10.1080/01621459.1952.10483441.

Li Q, Wacholder S, Hunter DJ, Hoover RN, Chanock S, Thomas G, Yu K (2009). "Genetic Background Comparison Using Distance-Based Regression, with Applications in Population Stratification Evaluation and Adjustment." *Genetic Epidemiology*, **33**(5), 432–441. doi:10.1002/gepi.20396.

Li Q, Yu K (2008). "Improved Correction for Population Stratification in Genome-Wide Association Studies by Identifying Hidden Population Structures." *Genetic Epidemiology*, **32**(3), 215–226. doi:10.1002/gepi.20296.

Li Q, Zheng G, Li Z, Yu K (2008). "Efficient Approximation of P-Value of the Maximum of Correlated Tests, with Applications to Genome-Wide Association Studies." *The Annals of Human Genetics*, **72**(3), 397–406. doi:10.1111/j.1469-1809.2008.00437.x.

Lin WY, Schaid DJ (2009). "Power Comparisons between Similarity-Based Multilocus Association Methods, Logistic Regression, and Score Tests for Haplotypes." *Genetic Epidemiology*, **33**(3), 183–197. doi:10.1002/gepi.20364.

Lin Y, Zhang M, Wang L, Pungpapong V, Fleet JC, Zhang D (2009). "Simultaneous Genome-Wide Association Studies of Anti-Cyclic Citrullinated Peptide in Rheumatoid Arthritis Using Penalized Orthogonal-Components Regression." *BMC Proceedings*, **3**(S7), S20. doi:10.1186/1753-6561-3-s7-s20.

Marchini J, Donnelly P, Cardon LR (2005). "Genome-Wide Strategies for Detecting Multiple Loci That Influence Complex Diseases." *Nature Genetics*, **37**(4), 413–417. doi:10.1038/ng1537.

Mardia KV, Kent JT, Bibby JM (2003). *Multivariate Analysis*. Academic Press, New York.

McArdle BH, Anderson MJ (2001). "Fitting Multivariate Models to Community Data: A Comment on Distance-Based Redundancy Analysis." *Ecology*, **82**(1), 290–297. doi:10.1890/0012-9658(2001)082[0290:fmmtcd]2.0.co;2.

Price AL, Patterson NJ, Plenge RM, Weinblatt ME, Shadick NA, Reich D (2006). "Principal Components Analysis Corrects for Stratification in Genome-Wide Association Studies." *Nature Genetics*, **38**(8), 904–909. doi:10.1038/ng1847.

Pritchard JK, Stephens M, Rosenberg NA, Donnelly P (2000). "Association Mapping in Structured Populations." *The American Journal of Human Genetics*, **67**(1), 170–181. doi:10.1086/302959.

R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Sasieni PD (1997). "From Genotypes to Genes: Doubling the Sample Size." *Biometrics*, **53**(4), 1253–1261. doi:10.2307/2533494.

SAS Institute Inc (2008). *SAS JMP Genomics, Version 3.2*. Cary. URL http://www.sas.com/.

Satten GA, Flanders WD, Yang Q (2001). "Accounting for Unmeasured Population Substructure in Case-Control Studies of Genetic Association Using a Novel Latent-Class Model." *The American Journal of Human Genetics*, **68**(2), 466–477. doi:10.1086/318195.

Shao J (2007). *Mathematical Statistics*. 2nd edition. Springer-Verlag. doi:10.1007/b97553.

Sladek R, Rocheleau G, Rung J, Dina C, Shen L, Serre D, Boutin P, Vincent D, Belisle A, Hadjadj S, *et al.* (2007). "A Genome-Wide Association Study Identifies Novel Risk Loci for Type 2 Diabetes." *Nature*, **445**(7130), 881–885. doi:10.1038/nature05616.

Terpstra TJ (1952). "The Asymptotic Normality and Consistency of Kendall's Test against Trend, When Ties Are Present in One Ranking." *Indagationes Mathematicae*, **14**(1952), 327–333. doi:10.1016/s1385-7258(52)50043-x.

Thoen MPM, Olivas NHD, Kloth KJ, Coolen S, Huang PP, Aarts MGM, Bac-Molenaar JA, Bakker J, Bouwmeester HJ, Broekgaarden C, Bucher J, Busscher-Lange J, Cheng X, Fradin EF, Jongsma MA, Julkowska MM, Keurentjes JJB, Ligterink W, Pieterse CMJ, Ruyter-Spira C, Smant G, Testerink C, Usadel B, van Loon JJA, van Pelt JA, van Schaik CC, van Wees SCM, Visser RGF, Voorrips R, Vosman B, Vreugdenhil D, Warmerdam S,

Wiegers GL, van Heerwaarden J, Kruijer W, van Eeuwijk FA, Dicke M (2017). "Genetic Architecture of Plant Stress Resistance: Multitrait Genome-Wide Association Mapping." *New Phytologist*, **213**(3), 1346–1362. doi:10.1111/nph.14220.

Tracy CA, Widom H (1994). "Level-Spacing Distributions and the Airy Kernel." *Communications in Mathematical Physics*, **159**(1), 151–174. doi:10.1007/bf02100489.

Wang L, Zhang W, Li Q, Zhu W (2015). ***AssocTests.data**: Data for the **AssocTests** Package*. R package version 0.0-1, URL https://github.com/statscueb/AssocTests.data.

Wang L, Zhang W, Li Q, Zhu W (2020). ***AssocTests**: Genetic Association Studies*. R package version 1.0-0, URL https://CRAN.R-project.org/package=AssocTests.

Wessel J, Schork NJ (2006). "Generalized Genomic Disance-Based Regression Methodology for Multilocus Association Analysis." *The American Journal of Human Genetics*, **79**(5), 792–806. doi:10.1086/508346.

Wickham H, Hester J, Chang W (2019). ***devtools**: Tools to Make Developing R Packages Easier*. R package version 2.2.1, URL https://CRAN.R-project.org/package=devtools.

Yu K, Li Q, Bergen AW, Pfeiffer RM, Rosenberg PS, Caporaso N, Kraft P, Chatterjee N (2009). "Pathway Analysis by Adaptive Combination of P Values." *Genetic Epidemiology*, **33**(8), 700–709. doi:10.1002/gepi.20422.

Yu K, Wang Z, Li Q, Wacholder S, Hunter DJ, Hoover RN, Chanock S, Thomas G (2008). "Population Substructure and Control Selection in Genome-Wide Association Studies." *PLOS One*, **3**(7), e2551. doi:10.1371/journal.pone.0002551.

Zang Y, Fung W, Zheng G (2010). ***Rassoc**: Robust Tests for Case-Control Genetic Association Studies*. R package version 1.0-3, URL https://CRAN.R-project.org/src/contrib/Archive/Rassoc/.

Zapala MA, Schork NJ (2006). "Multivariate Regression Analysis of Distance Matrices for Testing Associations between Gene Expression Patterns and Related Variables." *Proceedings of the National Academy of Sciences of the United States of America*, **103**(51), 19430–19435. doi:10.1073/pnas.0609333103.

Zaykin DV, Zhivotovsky LA, Westfall PH, Weir BS (2002). "Truncated Product Method for Combining P Values." *Genetic Epidemiology*, **22**(2), 170–185. doi:10.1002/gepi.0042.

Zhang W, Li Q (2015). "Nonparametric Risk and Nonparametric Odds in Quantitative Genetic Association Studies." *Scientific Reports*, **5**(12105). doi:10.1038/srep12105.

Zhao JH (2007). "**gap**: Genetic Analysis Package." *Journal of Statistical Software*, **23**(8), 1–18. doi:10.18637/jss.v023.i08.

Zheng G, Freidlin B, Gastwirth JL (2006). "Robust Genomic Control for Association Studies." *The American Journal of Human Genetics*, **78**(2), 350–356. doi:10.1086/500054.

Affiliation:

Lin Wang
School of Statistics
Capital University of Economics and Business
No. 121 Zhangjialukou,
Fengtai District, Beijing 100070, China
E-mail: wanglin2009@amss.ac.cn

Wei Zhang
LSC, Academy of Mathematics and Systems Science
Chinese Academy of Sciences
No. 55, Zhongguancun East Road,
Haidian District, Beijing 100190, China
E-mail: zhangwei@amss.ac.cn

Qizhai Li *(corresponding author)*
LSC, Academy of Mathematics and Systems Science
Chinese Academy of Sciences
No. 55, Zhongguancun East Road,
Haidian District, Beijing 100190, China
E-mail: liqz@amss.ac.cn
and
School of Mathematical Science
University of Chinese Academy of Sciences
Beijing 100049, China

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
June 2020, Volume 94, Issue 5 *Submitted:* 2017-10-26
doi:10.18637/jss.v094.i05 *Accepted:* 2019-01-16