metaCCA: Package for summary statistics-based multivariate meta-analysis of genome-wide association studies using canonical correlation analysis

Anna Cichonska

October 17, 2016

Contents

1 Introduction

A dominant approach to genome-wide association studies (GWAS) is to perform univariate tests between genotype-phenotype pairs. However, analysing related traits together results in increased statistical power and certain complex associations become detectable only when several variants are tested jointly. Currently, modest sample sizes of individual cohorts and restricted availability of individual-level genotype-phenotype data across the cohorts limit conducting multivariate tests. *metaCCA* allows to conduct multivariate analysis of a single or multiple GWAS based on univariate regression coefficients. It allows multivariate representation of both phenotype and genotype.

metaCCA extends the statistical technique of canonical correlation analysis to the setting where the original individual-level data are not available. Instead, *metaCCA* operates on three pieces of the full data covariance matrix: S_{XY} of univariate genotype-phenotype association results, S_{XX} of genotype-genotype correlations, and S_{YY} of phenotype-phenotype correlations. S_{XX} is estimated from a reference database matching the study population, e.g., the 1000 Genomes (www.1000genomes.org), and S_{YY} is estimated from S_{XY}.

This vignette explains how to use the `metaCCA` package. For more details about the method, see [1].

2 Input data

The package contains a simulated toy data set. Here, we will work with it to show an example of the meta-analysis of two studies using *metaCCA*. We will focus on the analysis of 10 SNPs and 10 traits (phenotypic variables). We will use univariate summary statistics across 1000 SNPs to estimate phenotypic correlation structures S_{YY} (here, correlations between 10 traits). You can have a look at the list of variables provided by typing:

```
library(metaCCA)
data( package = 'metaCCA' )
```

- **N1** - number of individuals in study 1.
- **N2** - number of individuals in study 2.
- **S_XY_full_study1** - univariate summary statistics of 10 traits across 1000 SNPs (study 1).
- **S_XY_full_study2** - univariate summary statistics of 10 traits across 1000 SNPs (study 2).
- **S_XY_study1** - univariate summary statistics of 10 traits across 10 SNPs (study 1).
- **S_XY_study2** - univariate summary statistics of 10 traits across 10 SNPs (study 2).
- **S_XX_study1** - correlations between 10 SNPs corresponding to the population underlying study 1.
- **S_XX_study2** - correlations between 10 SNPs corresponding to the population underlying study 2.

Tab-separated text files containing the data can be found in the `inst/extdata` folder (except **N1** and **N2** which are just numerical values). They could be read to R using **read.table** function with options **header=TRUE** and **row.names=1**.

```
# Number of individuals in study 1
print( N1 )

## [1] 1000

# Number of individuals in study 2
print( N2 )

## [1] 2000
```

In *metaCCA*, we consider the following two types of the multivariate association analysis.

- **Single-SNP–multi-trait analysis**
  One genetic variant tested for an association with a set of phenotypic variables (genotypic correlation structure S_{XX} not needed).
- **Multi-SNP–multi-trait analysis**
  A set of genetic variants tested for an association with a set of phenotypic variables.

2.1 Univariate summary statistics S_{XY}

Data frame **`S_XY`** with row names corresponding to SNP IDs (e.g., position or rs_id) and the following columns.

- **`allele_0`** - allele 0 (string composed of "A", "C", "G" or "T").
- **`allele_1`** - allele 1 (string composed of "A", "C", "G" or "T").
- Two columns for each trait to be included in the analysis:
  - **`traitID_b`** - univariate regression coefficients;
  - **`traitID_se`** - corresponding standard errors;

  ("traitID" in the column name must be an ID of a trait specified by a user. Do not use underscores "_" in trait IDs outside "_b"/"_se" in order for the IDs to be processed correctly.)

```
# Part of the S_XY data frame for study 1

print( head(S_XY_study1[,1:6]), digits = 3 )

##       allele_0 allele_1 trait1_b trait1_se trait2_b trait2_se
## rs10         G        T  -0.0196    0.0448  -0.0256    0.0449
## rs80         G        T   0.0624    0.0607   0.0595    0.0608
## rs140        A        C   0.0239    0.0432   0.0157    0.0433
## rs170        A        T   0.0214    0.0483   0.0136    0.0483
## rs172        A        T   0.0205    0.0481   0.0163    0.0481
## rs174        T        G   0.0187    0.0479   0.0143    0.0480
```

2.2 Genotypic correlation structure S_{XX}

Data frame **S_XX** containing correlations between SNPs. It is needed only in case of multi-SNP–multi-trait analysis. Row names (and, optionally, column names) must correspond to SNP IDs. You can estimate correlations between SNPs from a reference database matching the study population, e.g., the 1000 Genomes project (www.1000genomes.org).

```
# Part of the S_XX data frame for study 1

print( head(S_XX_study1[,1:6]), digits = 3 )

##          rs10   rs80   rs140 rs170  rs172  rs174
## rs10   1.0000 -0.465  0.0878 0.119  0.146  0.149
## rs80  -0.4646  1.000  0.3161 0.171 -0.221 -0.226
## rs140  0.0878  0.316  1.0000 0.536 -0.214 -0.216
## rs170  0.1187  0.171  0.5360 1.000  0.406  0.406
## rs172  0.1456 -0.221 -0.2142 0.406  1.000  0.998
## rs174  0.1490 -0.226 -0.2156 0.406  0.998  1.000
```

3 metaCCA - workflow

3.1 Estimation of phenotypic correlation structure S_{YY}

In *metaCCA*, correlations between traits are estimated from univariate summary statistics S_{XY}. Specifically, each entry of the phenotypic correlation matrix S_{YY} corresponds to a Pearson correlation between univariate regression coefficients of two phenotypic variables across genetic variants. The higher the number of genetic variants, the lower the error of the estimate. See [1] for more details.

Here, we will estimate correlations between 10 traits using **estimateSyy** function. In each case, we will use summary statistics of 1000 SNPs. However, in practice, summary statistics of at least one chromosome should be used in order to ensure good quality of S_{YY} estimate. **estimateSyy** can be used no matter if the univariate analysis has been performed on standardised data (meaning that the genotypes were standardised before regression coefficients and standard errors were computed) or non-standardised data.

The function takes one argument - **S_XY** - data frame with univariate summary statistics in the form described in section 2.1 of this vignette.

```
# Estimating phenotypic correlation structure of study 1
S_YY_study1 = estimateSyy( S_XY = S_XY_full_study1 )

# Estimating phenotypic correlation structure of study 2
S_YY_study2 = estimateSyy( S_XY = S_XY_full_study2 )
```

estimateSyy returns a matrix **S_YY** containing correlations between traits given as input; here, 10 traits. Let's display a part of the resulting matrix for study 1.

```
print( head(S_YY_study1[,1:6]), digits = 3 )

##        trait1 trait2 trait3 trait4 trait5 trait6
## trait1  1.000  0.995  0.912  0.991  0.977  0.949
## trait2  0.995  1.000  0.942  0.998  0.991  0.933
## trait3  0.912  0.942  1.000  0.955  0.977  0.807
## trait4  0.991  0.998  0.955  1.000  0.996  0.922
## trait5  0.977  0.991  0.977  0.996  1.000  0.898
## trait6  0.949  0.933  0.807  0.922  0.898  1.000
```

3.2 Genotype-phenotype association analysis

The package contains two functions for performing the association analysis:

- **metaCcaGp** - runs the analysis according to *metaCCA* algorithm;
- **metaCcaPlusGp** - runs the analysis according to a variant of *metaCCA*, namely *metaCCA+*, where the full covariance matrix is shrunk beyond the level guaranteeing its positive semidefinite property (see [1] for more details).

Both functions require the same inputs, and they have the same output format. They accept a varying number of inputs, depending on the type of the association analysis. Traits and SNPs included in the analysis must be the same for the studies that are meta-analysed together.

In the next step, we will perform a meta-analysis of two studies, where we will test single SNPs for an association with a group of 10 traits (single-SNP–multi-trait analysis). At the end, we will also analyse several SNPs jointly (multi-SNP–multi-trait analysis).

3.2.1 Single-SNP–multi-trait analysis

By default, **metaCcaGp** and **metaCcaPlusGp** perform single-SNP–multi-trait analysis, where each given SNP is analysed in turn against all given phenotypic variables. The required inputs are as follows.

- **nr_studies** - number of studies analysed.
- **S_XY** - a list of data frames (one for each study) with univariate summary statistics corresponding to SNPs and traits to be included in the analysis (in the form described in section 2.1);

- **std_info** - a vector with indicators (one for each study) if the univariate analysis has been performed on standardised (**1**) or non-standardised (**0**) data (most likely the data were not standardised - the genotypes were not standardised before univariate regression coefficients and standard errors were computed - option **0** should be used);
- **S_YY** - a list of matrices (one for each study), estimated using **estimateSyy** function, containing correlations between traits to be included in the analysis;
- **N** - a vector with numbers of individuals in each study.

We will first run the default single-SNP–multi-trait analysis of two studies using provided toy data. Each of 10 SNPs will be tested for an association with the group of 10 traits.

```
# Default single-SNP--multi-trait meta-analysis of 2 studies

# Association analysis according to metaCCA algorithm
metaCCA_res1 = metaCcaGp( nr_studies = 2,
                          S_XY = list( S_XY_study1, S_XY_study2 ),
                          std_info = c( 0, 0 ),
                          S_YY = list( S_YY_study1, S_YY_study2 ),
                          N = c( N1, N2) )

# Association analysis according to metaCCA+ algorithm
metaCCApl_res1 = metaCcaPlusGp( nr_studies = 2,
                                S_XY = list( S_XY_study1, S_XY_study2 ),
                                std_info = c( 0, 0 ),
                                S_YY = list( S_YY_study1, S_YY_study2 ),
                                N = c( N1, N2 ) )
```

The output is a data frame with row names corresponding to SNP IDs. There are two columns containing the following information for each analysed SNP:

- **r_1** - leading canonical correlation value,
- **-log10(p-val)** - p-value in the -log10 scale.

```r
# Result of metaCCA
print( metaCCA_res1, digits = 3 )
```

```
##          r_1 -log10(p-val)
## rs10  0.0448        0.0892
## rs80  0.0356        0.0196
## rs140 0.0770        1.2318
## rs170 0.0757        1.1538
## rs172 0.0496        0.1616
## rs174 0.0497        0.1624
## rs176 0.0717        0.9294
## rs178 0.0731        1.0070
## rs180 0.0709        0.8889
## rs200 0.0521        0.2109
```

```r
# Result of metaCCA+
print( metaCCApl_res1, digits = 3 )
```

```
##          r_1 -log10(p-val)
## rs10  0.0342       0.01467
## rs80  0.0306       0.00622
## rs140 0.0705       0.86827
## rs170 0.0685       0.77279
## rs172 0.0342       0.01472
## rs174 0.0344       0.01516
## rs176 0.0640       0.57622
## rs178 0.0471       0.11958
## rs180 0.0460       0.10501
## rs200 0.0382       0.03188
```

If you wish, you can also run the association analysis of only one selected SNP. In such case, two additional inputs need to be given:

- **`analysis_type`** - indicator of the analysis type: **`1`**;
- **`SNP_id`** - ID of the SNP of interest.

Let's run the analysis for a SNP with ID 'rs80'; it will be tested for an association with the group of 10 provided traits.

```r
# Single-SNP--multi-trait meta-analysis of 2 studies
# and one selected SNP

# metaCCA
metaCCA_res2 = metaCcaGp( nr_studies = 2,
                          S_XY = list( S_XY_study1, S_XY_study2 ),
                          std_info = c( 0, 0 ),
                          S_YY = list( S_YY_study1, S_YY_study2 ),
                          N = c( N1, N2 ),
                          analysis_type = 1,
                          SNP_id = 'rs80' )

# Result of metaCCA
print( metaCCA_res2, digits = 3 )

##         r_1 -log10(p-val)
## rs80 0.0356        0.0196
```

```r
# metaCCA+
metaCCApl_res2 = metaCcaPlusGp( nr_studies = 2,
                                S_XY = list( S_XY_study1, S_XY_study2 ),
                                std_info = c( 0, 0 ),
                                S_YY = list( S_YY_study1, S_YY_study2 ),
                                N = c( N1, N2 ),
                                analysis_type = 1,
                                SNP_id = 'rs80' )

# Result of metaCCA+
print( metaCCApl_res2, digits = 3 )

##         r_1 -log10(p-val)
## rs80 0.0306       0.00622
```

3.2.2 Multi-SNP–multi-trait analysis

In order to analyse multiple SNPs jointly, you need to provide the following additional inputs:

- **analysis_type** - indicator of the analysis type: **2**;
- **SNP_id** - a vector with IDs of SNPs to be analysed jointly;
- **S_XX** - a list of data frames (one for each study) containing correlations between SNPs to be analysed (in the form described in section 2.2).

Here, we will run the analysis of 5 SNPs with IDs 'rs10', 'rs80', 'rs140', rs170' and 'rs172'. They will be tested jointly for an association with the group of 10 traits.

```
# Multi-SNP--multi-trait meta-analysis of 2 studies

# metaCCA
metaCCA_res3 = metaCcaGp( nr_studies = 2,
                          S_XY = list( S_XY_study1, S_XY_study2 ),
                          std_info = c( 0, 0 ),
                          S_YY = list( S_YY_study1, S_YY_study2 ),
                          N = c( N1, N2 ),
                          analysis_type = 2,
                          SNP_id = c( 'rs10', 'rs80', 'rs140',
                                      'rs170', 'rs172' ),
                          S_XX = list( S_XX_study1, S_XX_study2 ) )

# Result of metaCCA
print( metaCCA_res3, digits = 3 )

##                                                r_1 -log10(p-val)
## c("rs10", "rs80", "rs140", "rs170", "rs172") 0.0872         0.141
```

```
# metaCCA+
metaCCApl_res3 = metaCcaPlusGp( nr_studies = 2,
                                S_XY = list( S_XY_study1, S_XY_study2 ),
                                std_info = c( 0, 0 ),
                                S_YY = list( S_YY_study1, S_YY_study2 ),
                                N = c( N1, N2 ),
                                analysis_type = 2,
                                SNP_id = c( 'rs10', 'rs80', 'rs140',
                                            'rs170', 'rs172' ),
                                S_XX = list( S_XX_study1, S_XX_study2 ))

# Result of metaCCA+
print( metaCCApl_res3, digits = 3 )

##                                              r_1 -log10(p-val)
## c("rs10", "rs80", "rs140", "rs170", "rs172") 0.079        0.0102
```

If all studies included in the meta-analysis have the same underlying population (e.g., Finnish), only one genotypic correlation structure is needed. Let's assume that this is the case for two studies in our example.

```
S_XX_common = S_XX_study1
```

Then, association analysis according to *metaCCA* and *metaCCA+* would be run as follows.

```
# metaCCA
metaCCA_res4 = metaCcaGp( nr_studies = 2,
                          S_XY = list( S_XY_study1, S_XY_study2 ),
                          std_info = c( 0, 0 ),
                          S_YY = list( S_YY_study1, S_YY_study2 ),
                          N = c( N1, N2 ),
                          analysis_type = 2,
                          SNP_id = c( 'rs10', 'rs80', 'rs140',
                                      'rs170', 'rs172' ),
                          S_XX = list( S_XX_common, S_XX_common ) )

# Result of metaCCA
print( metaCCA_res4, digits = 3 )

##                                                r_1 -log10(p-val)
## c("rs10", "rs80", "rs140", "rs170", "rs172") 0.0864         0.131
```

```
# metaCCA+
metaCCApl_res4 = metaCcaPlusGp( nr_studies = 2,
                                S_XY = list( S_XY_study1, S_XY_study2 ),
                                std_info = c( 0, 0 ),
                                S_YY = list( S_YY_study1, S_YY_study2 ),
                                N = c( N1, N2 ),
                                analysis_type = 2,
                                SNP_id = c( 'rs10', 'rs80', 'rs140',
                                            'rs170', 'rs172' ),
                                S_XX = list( S_XX_common, S_XX_common ))

# Result of metaCCA+
print( metaCCApl_res4, digits = 3 )

##                                                r_1 -log10(p-val)
## c("rs10", "rs80", "rs140", "rs170", "rs172") 0.0788       0.00974
```

4 Summary

In this vignette, we have followed the procedure for association testing between multivariate genotype and multivariate phenotype based on univariate summary statistics using *metaCCA* algorithm and its variant *metaCCA+*. We used a simulated data set to demonstrate an example of meta-analysis of two genome-wide association studies.

For more information on the method, see [1].

References

[1] A Cichonska, J Rousu, P Marttinen, AJ Kangas, P Soininen, T Lehtimäki, OT Raitakari, MR Järvelin, V Salomaa, M Ala-Korpela, S Ripatti, M Pirinen (2016) metaCCA: Summary statistics-based multivariate meta-analysis of genome-wide association studies using canonical correlation analysis. *Bioinformatics*, btw052 (in press, to be updated).