A flexible copula-based approach for the analysis of secondary phenotypes in ascertained samples

Working with SPAC

Karim Oualkacha, Geneviève Lefebvre, Fodé Tounkara & Celia M.T. Greenwood

November 28, 2019

Contents

1 Introduction

This is a unified copula-based framework which tests for secondary phenotypes association and a single SNP in presence of selection bias. The method fits both retrospective and prospective likelihoods to handel selection bias under case-control (CC), extreme-trait (ET) and mulitple-trait (MT) sampling designs. The dependence between primary and secondary phenotypes is modelled via copulas. Thus, SPAC is a robust method for non-normality assumption of the secondary trait.

SPAC is an R package that contains methods to perform genetic association of a quantitative secondary phenotype and a single SNP in presence of selection bias. SPAC is a unified copula-based framework fits both retrospective and prospective likelihoods to handel selection bias under case-control (CC), extreme-trait (ET) and multiple-trait (MT) sampling designs. The dependence between primary and secondary phenotypes is modelled via copulas. Thus, SPAC is a robust method for non-normality assumption of the secondary trait; improved power is possible by appropriate modelling of the primary-secondary phenotypes joint distribution via copula models.

The main user-visible function of the package is `SPAC()` function which can be used to analyse single-SNP/Secondary phenotype association in one go.

The main function allows for prospective and retrospective copula-based likelihoods to handle selection bias via the argument *method*:

- pros (default), prospective copula-based likelihood
- retros, retrospective copula-based likelihood

Of note, the retrospective method does not handle covariates while the prospective-based method controls for covariates. The retrospective-based method, *method="retros"*, is the computationally fastest method, and these p-values can be used to triage which SNPs of the genome should be re-analyzed with *method="pros"*.

SPAC allows also for primary-secondary dependence modelling using five copulas

- Gaussian, (default);
- Student, Student copula with degree of freedom equals 10;
- Clayton, Clayton copula;
- Gumbel, Gumbel copula;
- Frank, Frank copula.

In order to run any of the examples below, the package needs to be installed and loaded first, of course:

```
library(devtools)

## Loading required package:  usethis

#devtools::install_github('micau80/SPAC', ref = "micau80-patch-1", dep =FALSE)
                         #build_vignettes = TRUE, dep =FALSE)
devtools::install_github('micau80/SPAC', dep =FALSE)

## Skipping install of 'SPAC' from a github remote, the SHA1 (4c5bfaec)
has not changed since last install.
##  Use `force = TRUE` to force installation

library(SPAC)

## Loading required package:  MASS
## Loading required package:  LaplacesDemon
## Loading required package:  VineCopula
## Loading required package:  copula
##
## Attaching package:  'copula'
## The following object is masked from 'package:LaplacesDemon':
##
##    interval
```

2 Loading the input data

Before running an association test with one (or more) of the methods, the following data needs to be present[1]:

- *Phenotype data*; primary and secondary phenpotypes data should be present separately in the form of an R vector (one value for each individual). It is up to you (as user) to create the vector, for example by reading it from a CSV file using R's `read.csv()` function or like this:

```
  data(data, package = "SPAC")
# data is .RData file contains 3 examples of data sets for CC, ET and MT designs
# y1cc: primary trait under the case-control (CC) design
  head(y1cc)

## [1] 1 1 1 1 1 1

# y2cc: secondary trait under the case-control (CC) design
  head(y2cc)

## [1] 4.5444333 4.6088662 3.4706840 4.3158411 0.5313973 7.2174877
```

- *Covariate data*; this data should be present in the form of a matrix. Like the phenotype data it us up to you to load this data. For example:

```
  data(data, package = "SPAC")
# data.file is .RData file contains 3 examples of data sets for CC, ET and MT designs
# matrix of two confounders/covariates: one dichotmous and one continuous covariate
  dim(cov.matCC)

## [1] 1000    2

  head(cov.matCC)

##       conf.1     conf.2
## 79222      1  2.1270291
## 48922      0  2.1488196
## 5552       0  1.9549483
## 98072      1  1.4939171
## 3987       0 -0.5533521
## 3840       1  2.5436760
```

[1] The example code in this vignette uses files that are included in the SPAC package, that is why the `package` argument to the `system.file()` function is used.

- *Genotype data*; SNP genotype data can be in the form of an R vector (one value for each individual).

```
  data(data, package = "SPAC")
# data is .RData file contains 3 examples of data sets for CC, ET and MT designs
# SNP data for the CC design: a vector of legnth
  length(markerCC)

## [1] 1000

  head(markerCC)

## [1] 1 1 1 0 0 1
```

3 Analysing a single SNP

With the phenotype data, the covariates and the genotype data loaded it is time for tests of association.

3.1 Using SPAC for a case-control design

This is the simplest way to run SPAC for a single-SNP/secondary phenotype association test under the CC design:

```
SNPresults <- SPAC(y1 = y1cc,
                   y2 = y2cc,
                   G = markerCC,
                   covariates = as.matrix(cov.matCC),
                   link = "probit",
                   copfit = "Gaussian",
                   method = "pros",
                   Design = "CC",
                   prev = 0.1)

## Starting association analysis of the SNP...

SNPresults

## $intercept.SNP.SecP
## [1] 0.6383071 0.0703275
##
## $SNP.SecP
## [1] 0.01637375 0.05628824
##
## $P.value.SecP
## [1] 0.7711346
##
## $intercept.SNP.PrP
## [1] -2.24384294  0.09328811
##
## $SNP.PrP
## [1] 0.17561193 0.06694834
##
## $P.value.PrP
## [1] 0.00871347
##
## $theta
##
## 0.4744207
##
## $tau
##       tau
## 0.3146849
##
## $df2
##
## 7188.598
##
## $AIC
## [1] 3885.297
```

- Here the `prev` is a scalar between 0 and 1, specifies the primary phenotype prevalence. It is needed for the `method = "pros"` and `Design = "CC"` or `Design = "MT"`.

- The `link` is a character, specifies the link function to be used for modelling the marginal distribution of the binary primary phenotype for the CC and MT designs. The available link functions are `link=c("probit","logit","cloglog")`. The defaut is `link = "probit"`, which the liability latent model.

- The `copfit` is a character that selects the copula model to use for modelling primary-secondary phenptypes dependence. Can be one of the following:
  - `copfit = "Gaussian"`, (default)
  - `copfit = "Student"`, Student copula with degree of freedom equals 10
  - `copfit = "Clayton"`, Clayton copula
  - `copfit = "Gumbel"`, Gumbel copula
  - `copfit = "Frank"`, Frank copula

3.2 Using SPAC for an extrem-trait (ET) design

The next code run SPAC for a single-SNP/secondary phenotype association test under the ET design:

```r
SNPresults.ET <- SPAC(y1 = y1et,
                   y2 = y2et,
                   G = markeret,
                   covariates = as.matrix(cov.matCC),
                   copfit = "Gaussian",
                   method = "pros",
                   Design = "ET",
                   cutoffs = c(-1.498953,2.174215),
                   p.lu =c(0.1,0.1)
                   )

## Starting association analysis of the SNP...

SNPresults.ET

## $intercept.SNP.SecP
## [1] 0.39526153 0.08734849
##
## $SNP.SecP
## [1] 0.07040854
##
## $P.value.SecP
## [1] 0.5432109
##
## $intercept.SNP.PrP
## [1] -0.2373004  0.0560055
##
## $SNP.PrP
## [1] 0.08189576 0.04184267
##
## $P.value.PrP
## [1] 0.05032032
##
## $theta
##
## 0.7362122
##
## $tau
##       tau
## 0.5267749
##
## $df2
```

```
##
## 35.11171
##
## $AIC
## [1] 5201.318
```

- The `cutoffs` is a vector or scalar, depending on the sampling mechanism design `Design = "ET"` or `Design = "MT"`. For the ET design the `cutoffs` is a 2×1 vector, `cutoffs = c(ylb,yub)`, with
  - ylb is the lower primary trait threshold
  - yub is the upper primary trait threshold
- The `p.lu` is a vector specifying the proportion of individuals with lower and upper extreme primary trait. It is needed for the `method = "pros"` and `Design = "ET"`.

Of note, the ET design does not need a `link` argument since it fits a liability threshold model for the primary phenotype.

3.3 Using SPAC for an multiple-trait (MT) design

The next code run SPAC for a single-SNP/secondary phenotype association test under the MT design. Here we use `copfit = "Clayton"` as a copula model to model the joint distribution of the primary-secondary phenotypes:

```
SNPresults.MT <- SPAC(y1 = y1mt,
                   y2 = y2mt,
                   G = markermt,
                   covariates = as.matrix(cov.matMT),
                   link = "probit",
                   copfit = "Clayton",
                   method = "pros",
                   Design = "MT",
                   cutoffs = 1.865465,
                   prev = 0.1,
                   prev2 =0.06112)

## Starting association analysis of the SNP...
## Warning in sqrt(diag(mvar)):  production de NaN

SNPresults.MT

## $intercept.SNP.SecP
## [1] 0.6693276 0.0736944
##
```

```
## $SNP.SecP
## [1] 0.05814432
##
## $P.value.SecP
## [1] 0.6116386
##
## $intercept.SNP.PrP
## [1] -2.2577354  0.1104644
##
## $SNP.PrP
## [1] 0.15285477 0.06968403
##
## $P.value.PrP
## [1] 0.02826844
##
## $theta
##
## 1.225416
##
## $tau
##       tau
## 0.3799249
##
## $df2
##
## 50.97831
##
## $AIC
## [1] 3757.415
```

- In the MT design, the `cutoffs` is a scalar, `cutoffs = y2ub`, with `y2ub` is the upper secondary trait threshold
- The `prev2` is a scalar between 0 and 1, specifies the proportion of diseased individuals with secondary trait exceeding the threshold `y2ub`. It is needed for the `method ="pros"` and `Design = "MT"`.

References

[1] Fodé Tounkara, Geneviève Lefebvre, Celia MT Greenwood and Karim Oualkacha (2019). *A flexible copula-based approach for the analysis of secondary phenotypes in ascertained samples*. Statistics in Medicine. 1-32. Under revision.