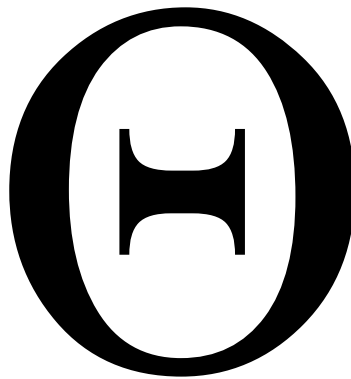


CoCoa documentation: software version 1.0



Author: Steven Maenhout
Affiliation: University College Ghent
E-mail: Steven.Maenhout@hogent.be
Website: <http://webs.hogent.be/cocoa>

Table of Contents

1 INTRODUCTION.....	3
2 BUILDING A PROJECT.....	4
2.1 INPUT FILE FORMAT.....	4
2.2 PROJECT.....	4
3 COANCESTRY ESTIMATORS.....	5
3.1 AIS.....	5
3.2 BNO.....	5
3.3 LOL.....	6
3.4 MLE.....	6
3.5 WAIS.....	6
4 MATRIX MANIPULATIONS.....	7
4.1 BENDING.....	7
4.1.1 <i>Spectral decomposition-based bending</i>	7
4.1.2 <i>MC bending</i>	7
4.2 BOUNDING.....	8
4.3 INVERSION.....	8
5 OUTPUT FILE FORMATS.....	8
6 REFERENCES.....	9

1 Introduction

The program CoCoo implements several marker-based coancestry estimators from the fields of population and quantitative genetics. The idea is to construct a symmetric matrix containing an estimator for the coefficient of coancestry for all pairs of genotyped individuals in the data set. Such a matrix is typically used for modeling the covariance between random additive genetic components when performing a linear mixed model analysis of phenotypical measurements obtained in breeding programs or association studies.

The coefficient of coancestry between two genotypes can be loosely defined as the probability that a randomly selected allele at a particular locus of one genotype is identical to a randomly selected allele at the same locus of the other genotype because of a common inheritance from a nearby ancestor. Exact pedigree information allows to calculate the expectation of this probability when it is assumed that at each generation both parents provide an equal contribution to their offspring's genotype and the latter does not experience selection of any kind. These conditions are usually only valid in natural populations which usually implies that there is no detailed pedigree information available. The assumptions are generally not met in breeding pools, rendering the available pedigree data useless for coancestry estimation.

In natural populations, the lack of pedigree data can be circumvented by estimating the coefficient of coancestry from genetic fingerprints, usually constructed from molecular marker scores. Most of the available estimation procedures however, assume that the individuals under study are randomly selected genotypes from a population under Hardy-Weinberg equilibrium. In breeding programs, however, this assumption is usually invalid which in turn means that the estimators from the field of population genetics are not able to guarantee statistically desirable properties like unbiasedness or minimum variance. In fact, even if there was a way to obtain an unbiased estimator of the coefficient of coancestry in breeding pools, the obtained estimate would be highly dependent on the analyst's subjective interpretation of a nearby common ancestor. As a result, breeders and researchers apply a wide variety of estimators with different inherent assumptions and different meanings of a 0 coefficient of coancestry. CoCoo allows for the estimation of the most commonly used marker-based coancestry estimators from the fields of population and quantitative genetics.

If a coancestry estimator is used for constructing the backbone of a covariance matrix involved in the variance structure of a linear mixed model, the estimated symmetric matrix of pairwise coancestries should, from a theoretical perspective, be at least positive semi-definite (psd). In fact, most if not all software packages for linear mixed model analysis require the matrix to be strictly positive definite, a property that can not be guaranteed by any marker-based coancestry estimation procedure. Certain estimators like AIS or WAIS (Maenhout et al., 2009) guarantee that the estimated coancestry matrix will be at least psd. Unfortunately, singular matrices can still occur for example when the number of genotyped markers is small compared to the number of genotyped individuals. All other marker-based coancestry estimation procedures can only guarantee that the the resulting coancestry matrix will be symmetrical. Therefore, if the estimated matrix turns out to be non-psd or singular, its entries can be slightly modified to obtain a positive definite matrix in a procedure named matrix bending. Bending an estimated coancestry matrix towards positive definiteness always comes at the cost of introducing a certain amount of bias to the coancestry estimates. This is generally the case for every bending strategy and the amount of bias will be highly dependent on the level of rank-deficiency of the initial matrix. In the extreme case, the

estimated matrix will be so ill-conditioned that it shares little or no resemblance to the the resulting positive definite matrix after bending. CoCoo implements two bending strategies, one based on spectral decomposition and the other uses a more time consuming Monte Carlo algorithm.

Besides marker-based estimation routines and matrix bending procedures, CoCoo provides several matrix manipulation tools that are often needed when dealing with coancestry matrices such as bounding matrix elements within the [0,1] interval or matrix inversion. The estimated matrices can be saved in several file formats that are used by the most common linear mixed model software packages such as ASReml, Wombat and SAS.

The interface of CoCoo was designed to resemble that of the program structure (<http://pritch.bsd.uchicago.edu/structure.html>), as most breeders and researchers involved in quantitative genetics are well acquainted with this software. The computational part of the program is performed by two applications named cocoacore and matrixmanip, both of which are written in C++. They rely on three open-source libraries: Newmat (<http://www.robertnz.net/>), OPT++ (<http://acts.nersc.gov/opt++/>) and GNU GSL (<http://www.gnu.org/software/gsl/>). These core components can be used as stand-alone, command-line applications. The graphical interface is written in Java and requires that a recent Java runtime Engine is installed. We distribute source code as well as executables for Windows and Linux. The CoCoo graphical user interface and its two core components are published under the GNU General Public License (GPL).

2 Building a project

2.1 Input file format

CoCoo uses the same input file format as the program structure. We therefore refer to the structure manual (<http://pritch.bsd.uchicago.edu/structure.html>) for the exact specifications of your input data file. Initially, all coancestry estimation routines expected diploid genotypes and co-dominant molecular marker scores but all but one routine were adapted to allow for polyploid genotypes and dominant marker scores as well. Two estimation routines use population information to infer the coefficient of coancestry. If this population information is not available (structure parameter POPDATA=0) than these two estimators will not be available from the CoCoo selection menu.

2.2 Project

CoCoo stores all settings and resulting matrices that result from a single input data file into a project. First you need to construct this input data file as described in the previous section. Now, click on Project→New. This opens up a wizard to import the data which should be fairly similar to the import wizard of structure. The data are copied from the specified input file into the work directory chosen for the project. The wizard consists of four frames:

1. Specify the project directory, project name, and input data file.

2. Specify the basic characteristics of the data file (number of individuals, ploidy of the data (enter '2' for diploid organisms), number of loci, and the value that is used to indicate missing data. If the ploidy is greater or smaller than 2, MLE estimation is not available. Click on “Show data file format” to get a summary of the lengths and number of lines in the data file.
3. (Rows) Specify which, if any, of the optional extra row data are present: row of marker names, row of inter-marker distances and a row of phase data after each individual. Also tick the “single line” box if data for each individual are stored in a single row, instead of in the standard format of two rows per individual.
4. (Columns) Specify which of the optional column data are there: Individual ID (LABEL), Population of origin (POPDATA) (needed for BNO and WAIS estimation), USEPOPINFO flag, phenotype data, other extra columns of data prior to the genotype data that should be ignored by CoCoa.

When you've finished these steps, you'll get a summary of the data format. If this summary looks correct, click on 'proceed'. CoCoa will now attempt to load the data file and create the new project.

3 Coancestry estimators

CoCoa uses abbreviations for each type of coancestry estimator. These abbreviations are identical to those used in the paper by Maenhout et al. (2009).

3.1 AIS

The AIS or A likeness In State estimator calculates the probability that the two alleles drawn at a random locus of each of two genotypes, are identical. As this identity is not necessarily caused by a shared inheritance from a common ancestor, this estimator tends to display an upward bias. Moreover, the estimator assumes linkage equilibrium between genotyped loci, which is generally not the case when dealing with selected genotypes. However, as indicated in Maenhout et al. (2009), if the breeding pool consists of highly selected inbred lines, this estimator performs rather well, especially when compared to several estimators from the field of population genetics. AIS is guaranteed to deliver a coancestry matrix that is at least psd. The matrix elements are estimated probabilities and therefore always lie within the unit interval. The estimation routine allows for polyploid genotypes and no population information is needed.

3.2 BNO

BNO is the estimator described by Bernardo (1993) for use in hybrid breeding programs. BNO expects the breeding pool to be divided in two or more heterotic groups for which it is assumed that genotypes belonging to different groups are unrelated (coefficient of coancestry of 0). Therefore, CoCoa requires a column in the data file that indicates to which population (heterotic group) each genotype belongs. Again, linkage equilibrium is assumed for the genotyped loci. The estimator

does not guarantee a psd matrix and coancestry estimates can be negative. The estimation routine allows for genotypes with higher ploidy states.

3.3 LOI

LOI is the estimator described by Loiselle et al (1995). Technically, LOI is a modified coefficient of correlation between allele frequencies and as such, not a probability of allele identity by descent from a common ancestor. The estimator has its origins in population genetics and admits to producing coefficients of coancestry either greater than 1 or smaller than 0. The inherent assumptions on which this estimator relies are unspecified. Despite the lack of theoretical foundations or extensive empirical testing, this estimator has been used quite frequently in association studies for modeling the covariance of the genetic background of related individuals. The estimator does not guarantee a psd matrix. The implemented routine allows for genotypes with higher ploidy states.

3.4 MLE

MLE refers to the Maximum Likelihood Estimator introduced by Thompson (1975). Just like LOI, this estimator has its origins in the field of population genetics but it has the extra advantage of allowing for inbred genotypes. CoCoo uses a quasi-Newton nonlinear interior point optimization routine to find the 8-dimensional solution vector that maximizes the likelihood function. The estimate of the coefficient of coancestry is subsequently obtained from this solution vector. MLE does not guarantee a psd coancestry matrix but all estimates lie within the unit interval. Again, it is assumed that the genotyped loci are in linkage equilibrium and that unbiased population allele frequencies can be obtained from the available sample of genotyped individuals. Currently, the implemented routine only works for diploid genotypes. The MLE-based estimation of the coancestry matrix for n genotypes requires $n(n+1)/2$ distinct optimizations of the likelihood function. Therefore, these calculations can become quite time consuming, especially if n is large.

3.5 WAIS

Just like BNO, the WAIS or Weighted Alikeness In State estimator has been especially developed for use in hybrid breeding programs (Maenhout et al., 2009). This means that the set of genotyped individuals must belong to two or more distinct heterotic groups and this population information must be available in the input file. Contrary to BNO, WAIS is guaranteed to deliver a psd coancestry matrix and all matrix elements lie within the unit interval. As all other estimators, WAIS implicitly relies on linkage equilibrium between genotyped loci. The implemented routine allows for genotypes with a ploidy state higher than 2.

4 Matrix manipulations

4.1 Bending

CoCoa can perform matrix bending to allow inversion of ill-conditioned matrices. However, the provided matrix bending procedures should not be considered as magical solutions to singularity problems caused by, for example, using uninformative or small marker sets on a large numbers of genotypes. The more ill-conditioned the initial matrix is, the larger the bias will be that is introduced by the bending procedure.

4.1.1 Spectral decomposition-based bending

This bending procedure is based on an eigenvalue analysis of the input matrix where all negative or zero eigenvalues are replaced by a small positive value. The input matrix is then reconstructed from the original eigenvectors and these modified eigenvalues. The size of the small positive value is determined by the user's requested (l2 norm) condition number. This number expresses how numerically well-conditioned your bended matrix should be. Large numbers indicate a near singularity condition which in turns means that it is impossible (in fact not impossible, just unwise) to calculate the matrix inverse as it will be highly sensitive to small perturbations in your input matrix (usually caused by round-off errors due to the finite precision of floating point arithmetic). A small condition number on the other hand indicates that your matrix is well-conditioned and can therefore be inverted without further worry. To get an idea of the condition number of your matrix before bending, try inverting it first, as CoCoa's matrix inversion routine will give you the condition number as additional information. There are no strict rules on how small the condition number of a matrix should be before inversion can be considered. By default, the size of the matrix is used as the target condition number but this setting might introduce too much bias. To evaluate how much bias was introduced by bending, CoCoa provides the elementwise average distance between the original and the bended matrix. This criterion is obtained by calculating for each of the $n(n+1)/2$ unique elements in a coancestry matrix of size n , the distance (absolute value of the difference) between its value in the input and its value in the output matrix and consequently taking the average of these differences.

Although this spectral decomposition-based bending procedure is computationally fairly straightforward, there is no way to force each coancestry estimate within the unit interval. This means that after bending has been performed, it is most likely that several matrix elements will be negative or greater than 1, which excludes their interpretation as a probability. If the elements of the bent coancestry matrix must all lie in the unit interval, the MC bending procedure should be used instead.

4.1.2 MC bending

CoCoa can perform matrix bending by means of a Monte Carlo routine inspired by the bending routine implemented in the program FLBEND (Henshall and Meyer, 2002). This MC bending algorithm is computationally intensive and this is especially pronounced for larger matrices. CoCoa

therefore allows to set a maximum calculation time. The default is set to 120 seconds but if computation time is not important, you can leave this blank and set a maximum allowed average distance between the input and the bent matrix. The routine will then search for the nearest matrix that is positive definite. Although this bending routine is computationally quite demanding compared to spectral decomposition-based bending, it provides the option to restrict the matrix elements within the unit interval (or any other interval for that matter).

4.2 Bounding

CoCoo allows to bound the elements of an estimated matrix within a certain interval. Usually, you will want your coancestry estimators to lie within the unit interval. In this case, all elements that are greater than 1 are set equal to one and negative elements are set equal to 0.

4.3 Inversion

CoCoo provides a matrix inversion routine based on the singular value decomposition. Inverting an estimated coancestry matrix only makes sense if the matrix is well-conditioned and therefore has a small condition number. CoCoo will not perform the inversion if the reciprocal of the matrix condition number is smaller than the machine tolerance. In this case, this near singularity condition needs to be resolved for example by genotyping an extra set of markers. You can also bend the matrix to obtain a smaller condition number but as mentioned before, this introduces (additional) bias into your coancestry estimates.

5 Output file formats

CoCoo allows you to store the obtained coancestry matrices and their manipulated counterparts in various file formats. Most linear mixed model software packages that allow you to import a user-defined variance matrix require the matrix (usually its inverse) to be available in a separate text file. This text file should have a coordinate format, meaning that each line should contain a row index, a column index and the matrix element at that position. There are however subtle differences in the exact syntax between the different linear mixed model packages so CoCoo provides specific output formats for:

- ASReml
- Wombat
- SAS

Besides coordinate formats, CoCoo also allows you to export the matrix in a dense format. Each line in this text file is the equivalent of a row in the matrix. Elements belonging to different columns are separated by spaces. It is usually straight forward to import this file format in other programs or programming languages. As a coancestry matrix is symmetrical, you can also export the upper or lower triangle (including the main diagonal) instead of the full matrix.

6 References

- Bernardo R (1994) Prediction of maize single-cross performance using RFLPs and information from related hybrids. *Crop Sci* 34: 20–25
- Henshall JM, Meyer K (2002) PDMATRIX–Programs to make matrices positive definite. *Proceedings of the 7th World Congress on Genetics Applied to Livestock Production. Communication No. 28-12. Vol. 33: 753–754*
- Loiselle BA, Sork VL, Nason J, Graham C (1995) Spatial genetic structure of a tropical understory shrub, *Psychotria officinalis* (Rubiaceae). *Am J Bot* 82: 1420–1425
- Maenhout S, De Baets B, Haesaert G (2009) Marker-based estimation of the coefficient of coancestry in hybrid breeding programmes. *Theor Appl Genet* 118: 1181—1192
- Thompson EA (1975) The estimation of pairwise relationships. *Ann Hum Genet* 39:173–188