Scripts for generating synthetic genomic data.

These are scripts that can be used for generating synthetic data; specifically for evaluating genomic analysis scripts such as the genomic-relatedness method (**MVMLE** repository) and compressed sensing (**CS** repository). There are two general classes of synthetic data: 1) measurement matrix and 2) the response vector. The measurement matrix is meant to simulate the matrix with columns representing the genetic variants and rows that are the samples across subjects. The response vector is meant to model a phenotype.

edited: 04-18-14 by Shashaank Vattikuti

1 measurement matrix models

Binomial, independent measures with given maf

This generates a matrix of "genotypes" by sampling a Binomial distribution twice to give elements $\in \{0, 1, 2\}$. The distribution of each column is given by a user defined minor allele frequency (maf). This is generated by calling

G=mm_B(m,n,miss,miss_value,maf)

where m=number of subjects, n=number of genotypes, $miss$=the missing genotype rate, $miss\_value$ is the value to use for missing genotypes, and maf is a $\mathbb{R}^{n\times 1}$ vector of minor allele frequencies.

Normally distributed measures with given LD structure

This generates a matrix of "genotypes" by sampling a Normal distribution and imposing an LD (correlation) structure by projecting against the upper triangular matrix of the Cholesky decomposition for a given correlation matrix. This is generated by calling

G=mm_N(m,n,miss,miss_value,$\mathbf{R}$)

where m=number of subjects, n=number of genotypes, $miss$=the missing genotype rate, $miss\_value$ is the value to use for missing genotypes, and $\mathbf{R}$ is the $\mathbb{R}^{n\times n}$ correlation matrix across measurements (e.g., genetic variants, SNPs).

2 response vector models

Note that the measurement matrix ($\mathbf{G}$) used to generate the synthetic models does not need to come from above and can be used with real GWAS matrices as used in ref. [1].

univariate continuous model

A univariate continuous "phenotype" vector is generated based on the additive model,

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}, \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^{m\times 1}$ is the vector of phenotypes, $\mathbf{A} \in \mathbb{R}^{m\times n}$ is the matrix of standardized genotypes, $\mathbf{x} \in \mathbb{R}^{n\times 1}$ is the vector of partial regression coefficients, and $\mathbf{e} \in \mathbb{R}^{m\times 1}$ is the vector of residuals. This is implemented by

[x,y]=y_additive(G,miss_value,h2,xs,xtype,xsign,xmaf)

where $\mathbf{G}$ is the $m \times n$ genomic measurement matrix, $miss\_value$ is the missing genotype indicator value, and $h2$ is the (user defined) ratio of the variance due to additive "genetic" factors over the phenotype variance. The remaining arguments define the coefficient vector $\mathbf{x}$. The parameter xs are the number of nonzero coefficients (called nonzeros). The argument $xtype$ is a string from the set $\{$'Uniform', 'Hyperexponential1', 'Hyperexponential2'$\}$. Uniform defines nonzeros whose absolute values are equal. Hyperexponential1 and Hyperexponential2 are mixture of exponentials; specifically the sum of two exponentials with equal amplitude

and different decay constants. The decay constants for Hyperexponential1 are 0.05s and n and those of Hyperexponential2 are 0.2s and n, where s=number of nonzeros and n=number of coefficients. The argument $xsign$ is a string from the set {'pos', 'neg', 'posneg'} and defines whether the nonzeros are all positive or negative signed, or an equal mixture of both respectively. The argument $xmaf$ is a string from the set {'maf_low', 'maf_high', 'maf_high'} and defines whether the nonzeros are located on the low- or high- end, or randomly represent the maf spectrum for the given Binomial genomic measures (the script calculates the maf). This is an optional argument and is only employed if the matrix $\mathbf{G}$ is from a Binomial distribution.

3 Other notes

The example scripts in the **CS** repository call these functions. To run these examples, download the **GD** repository and add it to the MATLAB path. These also reference the correlation structure and MAF of chromosome 22 based on GENEVA-ARIC European-American data [1] located in the **GD** repository.The MAF vector is in the file *chr22_maf.mat* and can be directly loaded. Due to GitHub file constraints the correlations are package across two files called *chr22_SNPcorrelations_**. The correlation matrix can be unpacked by calling the function

unpack_chr22_SNPcorrelations(GDpath)

where *GDpath* is the path for the local **GD** repository (see the **CS** example *example2_samplesize_scan.m*).

References

[1] Vattikuti, S., Lee, J., Chang, C., Hsu, S., Chow, C.: Applying compressed sensing to genome-wide association studies. GigaScience (in review)