

GGEE Vignette

Virginie Stanislas

2016-03-03

1) Introduction

The GGEE package implements the “Gene-Gene Eigen Epistasis” approach to detect epistasis at the gene level in genome wide association studies (GWAS). This approach compute interaction variables for each gene pair then uses a penalized regression method based on group lasso to select the significant main or interaction effects.

The two main functions of this package are `BuiltEpiVar` and `GLmodel`. `BuiltEpiVar` allows to calculate interaction variables under four different interaction variable modeling approaches. The Eigen-epistasis approach find for each gene pair a component defined as the linear combination of gene markers (SNPs) having the highest correlation with the phenotype. The three other modeling approaches are inspired by previous literature proposals, they compute interaction variable using: Principal Component Analysis (PCA), Partial Least squares (PLS) or Canonical-Correlation Analysis (CCA). `GLmodel` fits a group lasso model on the genetic data set enhanced by interaction variables then uses a screen and clean procedure in order to compute p-values for each group. A group is either made with the SNPs from a given gene or of interaction terms relative to a given gene-pair interaction.

Additionally, the package allows to generate genotype and phenotype data under two phenotypic models.

2) Generating genotype and phenotype data

The GGEE package allows to generate gene structured data and associated continuous phenotype according to the model :

$$\mathbf{y} = \mathbf{X}^T \boldsymbol{\beta} + \mathbf{Z}^T \boldsymbol{\gamma} + \boldsymbol{\epsilon}$$

Where $\mathbf{y} \in \mathbb{R}^n$ denotes the vector of trait values for n individuals, $\mathbf{X} \in \{1, 2, 3\}^{n \times p}$ represents the SNP matrix, \mathbf{Z} the matrix gathering interaction variables and $\boldsymbol{\epsilon} \in \mathbb{R}^n$ a gaussian error term. The columns of \mathbf{X} are structured on G non overlapping genes. Each gene is described by a given number of SNPs p_g where $\sum_g p_g = p$. The matrix of interaction \mathbf{Z} is structured into $G(G-1)/2$ submatrices each submatrice being the group of interaction variables for a specific pair of genes.

The two functions `simGeno` and `simPheno` allows to respectively simulate genotype and phenotype data.

```
library(GGEE)
sizeGenesMain <- rep(6,2) # 2 genes with 6 SNPs
sizeGenesPair <- rep(6,2) # 2 genes with 6 SNPs
sizeGenesRemain <- rep(6,4) # 4 genes with 6 SNPs
SameMainPair <- FALSE # Specify that genes with interaction effects will not have main effects
N<- 600
causalSNPnb <- 2
corr <- 0.8
MAFcausalSNP=0.2

Geno <- simGeno(N=N, corr=corr, sizeGenesMain=sizeGenesMain, sizeGenesPair=sizeGenesPair,
sizeGenesRemain=sizeGenesRemain, SameMainPair=SameMainPair, MAFcausalSNP=MAFcausalSNP,
causalSNPnb=causalSNPnb)
```

With these parameters the function `simGeno` simulate a data set of 6 genes, each one composed of 6 SNPs, for 600 individuals. The 2 first genes are considered to have main effects and the gene 3 and gene 4 to have an interaction effect. For the four causal genes their 2 first SNPs are considered as causal variants. Rather than a defined number of causal SNPs by causal gene, it is possible to use a portion of causal SNPs with the option `causalSNPportion`. In this case the option `causalSNPnb` has to be `NULL`. In both cases, the SNPs considered as causal are the first listed in the gene. The MAF of each SNP is randomly set between the values `minMAF` and `maxMaf` (by default `minMAF=0.05` and `maxMaf=0.5`). For the causal SNPs the MAF value correspond to `MAFcausalSNP`. The correlation between SNPs belonging to the same gene is set to `corr=0.8`.

The output of `Geno` contain the following elements :

- The 600×48 genotype matrix X
- The list `listGenesSNP` that indicate the names of the SNPs composing each gene
- The vectors `MainEff` and `GenePair` which give the names of genes having respectively main or interaction effects. The size of the vector `GenePair` is an even number, the pairs being defined with genes successively taken two by two along the vector.
- The vector `MAF` which give the minor allele frequency observed for each simulated SNP

```
Geno$X[1:5,1:8]
```

```
##      Gene.1.SNP.1 Gene.1.SNP.2 Gene.1.SNP.3 Gene.1.SNP.4 Gene.1.SNP.5
## [1,]           1           1           1           1           1
## [2,]           1           1           1           1           1
## [3,]           1           1           1           1           1
## [4,]           1           1           1           1           1
## [5,]           1           1           1           1           1
##      Gene.1.SNP.6 Gene.2.SNP.1 Gene.2.SNP.2
## [1,]           2           1           1
## [2,]           1           1           2
## [3,]           1           1           1
## [4,]           2           2           1
## [5,]           1           1           1
```

```
Geno$listGenesSNP
```

```
## $Genes1
## [1] "Gene.1.SNP.1" "Gene.1.SNP.2" "Gene.1.SNP.3" "Gene.1.SNP.4"
## [5] "Gene.1.SNP.5" "Gene.1.SNP.6"
##
## $Genes2
## [1] "Gene.2.SNP.1" "Gene.2.SNP.2" "Gene.2.SNP.3" "Gene.2.SNP.4"
## [5] "Gene.2.SNP.5" "Gene.2.SNP.6"
##
## $Genes3
## [1] "Gene.3.SNP.1" "Gene.3.SNP.2" "Gene.3.SNP.3" "Gene.3.SNP.4"
## [5] "Gene.3.SNP.5" "Gene.3.SNP.6"
##
## $Genes4
## [1] "Gene.4.SNP.1" "Gene.4.SNP.2" "Gene.4.SNP.3" "Gene.4.SNP.4"
## [5] "Gene.4.SNP.5" "Gene.4.SNP.6"
##
## $Genes5
## [1] "Gene.5.SNP.1" "Gene.5.SNP.2" "Gene.5.SNP.3" "Gene.5.SNP.4"
```

```
## [5] "Gene.5.SNP.5" "Gene.5.SNP.6"
##
## $Genes6
## [1] "Gene.6.SNP.1" "Gene.6.SNP.2" "Gene.6.SNP.3" "Gene.6.SNP.4"
## [5] "Gene.6.SNP.5" "Gene.6.SNP.6"
##
## $Genes7
## [1] "Gene.7.SNP.1" "Gene.7.SNP.2" "Gene.7.SNP.3" "Gene.7.SNP.4"
## [5] "Gene.7.SNP.5" "Gene.7.SNP.6"
##
## $Genes8
## [1] "Gene.8.SNP.1" "Gene.8.SNP.2" "Gene.8.SNP.3" "Gene.8.SNP.4"
## [5] "Gene.8.SNP.5" "Gene.8.SNP.6"
```

```
Geno$MainEff
```

```
## [1] "Genes1" "Genes2"
```

```
Geno$GenePair
```

```
## [1] "Genes3" "Genes4"
```

```
Geno$MAF
```

```
## Gene.1.SNP.1 Gene.1.SNP.2 Gene.1.SNP.3 Gene.1.SNP.4 Gene.1.SNP.5
## 0.19666667 0.19916667 0.17416667 0.26500000 0.05416667
## Gene.1.SNP.6 Gene.2.SNP.1 Gene.2.SNP.2 Gene.2.SNP.3 Gene.2.SNP.4
## 0.26166667 0.20583333 0.19500000 0.18583333 0.22500000
## Gene.2.SNP.5 Gene.2.SNP.6 Gene.3.SNP.1 Gene.3.SNP.2 Gene.3.SNP.3
## NA 0.47250000 0.19750000 0.21083333 0.29333333
## Gene.3.SNP.4 Gene.3.SNP.5 Gene.3.SNP.6 Gene.4.SNP.1 Gene.4.SNP.2
## 0.47750000 0.35666667 0.13916667 0.19750000 0.21583333
## Gene.4.SNP.3 Gene.4.SNP.4 Gene.4.SNP.5 Gene.4.SNP.6 Gene.5.SNP.1
## 0.35000000 0.40250000 0.27916667 0.20416667 0.20333333
## Gene.5.SNP.2 Gene.5.SNP.3 Gene.5.SNP.4 Gene.5.SNP.5 Gene.5.SNP.6
## 0.20083333 0.14750000 0.11583333 0.24916667 0.45000000
## Gene.6.SNP.1 Gene.6.SNP.2 Gene.6.SNP.3 Gene.6.SNP.4 Gene.6.SNP.5
## 0.19333333 0.19333333 0.45083333 0.09500000 0.49666667
## Gene.6.SNP.6 Gene.7.SNP.1 Gene.7.SNP.2 Gene.7.SNP.3 Gene.7.SNP.4
## 0.40750000 0.19416667 0.20500000 0.44333333 0.26083333
## Gene.7.SNP.5 Gene.7.SNP.6 Gene.8.SNP.1 Gene.8.SNP.2 Gene.8.SNP.3
## 0.45416667 0.42583333 0.19916667 0.19500000 0.41750000
## Gene.8.SNP.4 Gene.8.SNP.5 Gene.8.SNP.6
## 0.18416667 0.40666667 0.43333333
```

Once the genotype matrix obtained, phenotype values can be simulated through the function `simPheno`. The function takes as parameters:

- the outputs of the `simGeno` function,
- two vectors of possible values for coefficients β and γ ,
- the number or portion of causal SNPs to consider by gene (It has to be the same value than the one chosen for `simGeno`),

- a r^2 value that calibrate the difficulty of the problem,
- a value for the intercept β_0 (default $\beta_0 = 0$),
- the model to consider to simulate interaction effects:
 - "SNPproduct": $Y_i = \beta_0 + \sum_g \beta_g \left(\sum_{k \in \mathcal{C}} X_{ik}^g \right) + \sum_{rs} \gamma_{rs} \left(\sum_{(j,k) \in \mathcal{C}^2} X_{ij}^r X_{ik}^s \right) + \epsilon_i$
 - "PCproduct": $Y_i = \beta_0 + \sum_g \beta_g \left(\sum_{k \in \mathcal{C}} X_{ik}^g \right) + \sum_{rs} \gamma_{rs} C_{i1}^r C_{i1}^s + \epsilon_i$

where \mathcal{C} and \mathcal{C}^2 are respectively the set of causal SNPs and causal interactions, and ϵ_i a random Gaussian variable. For each causal gene g a coefficient β_g is assigned to the standardized sum of the causal SNPs. for the interactions, in the first model "SNPproduct", all the causal SNPs from a causal pair (r, s) are pairwise multiplied and the interaction of the causal pair is represented by the standardized sum of the products. In the second model "PCproduct", the interaction is represented by the standardized product of the first PCA component $C_{.1}^r$ of gene r and the first PCA component $C_{.1}^s$ of gene s . The computation of PCA components is realized on the whole gene and not only on the causal SNPs.

```
# possible values for coef Beta or Gamma
pvBeta <- c(2,2)
pvGamma <- c(2,2)
r2 <- 0.4

Pheno <- simPheno(X=Geno$X, listGenes=Geno$listGenesSNP, MainEff=Geno$MainEff, GenePair=Geno$GenePair,
  model="SNPproduct", pvBeta=pvBeta, pvGamma=pvGamma, r2=r2, causalSNPnb=causalSNPnb)
```

The outputs of the function `simPheno` includes

- the vector of phenotype continuous values y ,
- the matrix G of the simulated main effects, each column represent one causal gene and correspond to the standardized sum of its causal SNPs,
- the matrix GG of the simulated interaction effects, each column represent one causal interaction defined depending of the selected model,
- values for the coefficient of determination R^2 when considering the model containing only simulated interaction effects R2I or only simulated main effects R2S or both simulated main and interaction effects R2T,
- a list `caract` with the characteristic of the simulation. The information about the part of the coefficient of determination R^2 hat can be attributed to either interaction effects $p_{R_I^2} = \frac{R_I^2}{R_T^2}$ or main effects

$p_{R_M^2} = \frac{R_M^2}{R_T^2}$ is given.

```
head(Pheno$y)
```

```
##           [,1]
## V1 -6.2029755
## V2  4.8071671
## V3  3.1516677
## V4  3.0961886
## V5  3.0868815
## V6  0.1770048
```

```
head(Pheno$G)
```

```
##          Genes1      Genes2
## V1 -0.7851669 -0.7974191
## V2 -0.7851669  0.1972825
## V3 -0.7851669 -0.7974191
## V4 -0.7851669  0.1972825
## V5 -0.7851669 -0.7974191
## V6 -0.7851669 -0.7974191
```

```
head(Pheno$GG)
```

```
##          X.Genes3.Genes4
## [1,]      -0.454578817
## [2,]       2.266092409
## [3,]       0.905756796
## [4,]      -0.908024022
## [5,]      -0.908024022
## [6,]      -0.001133613
```

```
Pheno[c("R2T", "R2I", "R2S")]
```

```
## $R2T
## [1] 0.4288422
##
## $R2I
## [1] 0.1334254
##
## $R2S
## [1] 0.3008971
```

```
Pheno$caract
```

```
## $MainEff
## [1] "Genes1" "Genes2"
##
## $nbSNPbyMainEff
## Genes1 Genes2
##      6      6
##
## $Coef_MainEff
## Genes1 Genes2
##      2      2
##
## $causalSNPMainEff
##          Genes1      Genes2
## [1,] "Gene.1.SNP.1" "Gene.2.SNP.1"
## [2,] "Gene.1.SNP.2" "Gene.2.SNP.2"
##
## $GenePair
## [1] "Genes3" "Genes4"
##
## $nbSNPbyInterGene
## Genes3 Genes4
```

```
##      6      6
##
## $Coef_GenePair
## X.Genes3.Genes4
##      2
##
## $causalSNPIter
##      Genes3      Genes4
## [1,] "Gene.3.SNP.1" "Gene.4.SNP.1"
## [2,] "Gene.3.SNP.2" "Gene.4.SNP.2"
##
## $beta0
## [1] 0
##
## $r2
## [1] 0.4
##
## $causalSNPportion
## NULL
##
## $causalSNPnb
## [1] 2
##
## $R2T
## [1] 0.4288422
##
## $Partr2I
## [1] 31.11293
##
## $Partr2S
## [1] 70.16498
```

3) The G-GEE method

Once genotype and phenotype data are obtained we can apply the G-GEE approach to seek for interaction effects. The first step is to create interaction variables from each gene couple, the second is to test for potential main or interaction effects.

Interaction variable modeling can be done with the function `BuiltEpiVar`. The function takes as parameters the matrix of genotype X , the vector of phenotypic traits y , a list `listGenesSNP` that indicate the names of the SNPs composing each gene and `nbcomp` the number of components to consider to compute interaction variables. Four different methods can be use to create interaction variables :

- "GGEE" which find for each gene pair its Eigen-epistasis Component that maximize the correlation between all possible SNP-SNP interactions and the phenotype.
- "PCA" which first compute PCA on each gene of the pair and represent the interaction with component products.
- "PLS" interaction variables are defined by components that maximize the covariance between the two genes and the phenotype.
- "CCA" interaction variables are here represent by the product of pairwise components obtained by a canonical correlation analysis on the gene pair.

Here we show an example using "GGEE"option. As this method can compute only one interaction by gene couple, the parameter `nbcomp` doesn't need to be used.

```
Int <- BuiltEpiVar(Geno$X, Pheno$y, method="GGE", listGenesSNP=Geno$listGenesSNP)
```

```
## [1] "X.Genes1.Genes2"
## [1] "X.Genes1.Genes3"
## [1] "X.Genes1.Genes4"
## [1] "X.Genes1.Genes5"
## [1] "X.Genes1.Genes6"
## [1] "X.Genes1.Genes7"
## [1] "X.Genes1.Genes8"
## [1] "X.Genes2.Genes3"
## [1] "X.Genes2.Genes4"
## [1] "X.Genes2.Genes5"
## [1] "X.Genes2.Genes6"
## [1] "X.Genes2.Genes7"
## [1] "X.Genes2.Genes8"
## [1] "X.Genes3.Genes4"
## [1] "X.Genes3.Genes5"
## [1] "X.Genes3.Genes6"
## [1] "X.Genes3.Genes7"
## [1] "X.Genes3.Genes8"
## [1] "X.Genes4.Genes5"
## [1] "X.Genes4.Genes6"
## [1] "X.Genes4.Genes7"
## [1] "X.Genes4.Genes8"
## [1] "X.Genes5.Genes6"
## [1] "X.Genes5.Genes7"
## [1] "X.Genes5.Genes8"
## [1] "X.Genes6.Genes7"
## [1] "X.Genes6.Genes8"
## [1] "X.Genes7.Genes8"
```

Int is a list composed of the interaction variable matrix `XBet` and a vector `interLength` indicating the number of interaction variables for each couple.

```
head(Int$XBet)
```

```
##      X.Genes1.Genes2 X.Genes1.Genes3 X.Genes1.Genes4 X.Genes1.Genes5
## [1,]      -0.37483330      -0.94670703      -0.38919401      -0.08003515
## [2,]      -0.39592814       0.91481607      -0.45494578      -0.92748635
## [3,]      -1.13993875      -0.62887479      -0.08627101      -0.92748635
## [4,]       0.07089144      -0.77947346      -0.44337368      -0.19737197
## [5,]      -0.65541149      -1.09900054      -0.98329433      -1.06206221
## [6,]      -0.98275928      -0.09209224      -1.12114191      -0.92748635
##      X.Genes1.Genes6 X.Genes1.Genes7 X.Genes1.Genes8 X.Genes2.Genes3
## [1,]      -0.9840520014      -1.04211751       0.82969437      -0.74267166
## [2,]      -1.1604971995      -0.09719768      -1.14936788       2.63455984
## [3,]      -0.5770498185      -1.03724033      -1.14936788      -0.70647238
## [4,]      -0.4921236289      -1.04211751      -0.08545953      -0.20605273
## [5,]      -0.0006838417      -1.19063609      -0.86384571      -0.77196252
## [6,]      -0.7251952842      -0.59579760      -0.42786058       0.07351239
##      X.Genes2.Genes4 X.Genes2.Genes5 X.Genes2.Genes6 X.Genes2.Genes7
## [1,]      -0.09313049       0.2318984      -0.7985800      -0.8502372
```

```
## [2,] 0.45555707 -0.2437522 -0.5602747 1.0355134
## [3,] -0.17521410 -0.9946540 -0.6498773 -1.0609108
## [4,] 0.28202240 0.6780865 0.2360542 -0.5501733
## [5,] -0.60399652 -0.7052613 0.7138565 -0.8725055
## [6,] -1.04522407 -0.8485122 -0.6000253 -0.4724529
## X.Genes2.Genes8 X.Genes3.Genes4 X.Genes3.Genes5 X.Genes3.Genes6
## [1,] 1.2861504 -0.7459509 -0.53651266 -1.22455631
## [2,] -0.6156480 2.2484680 1.12889685 0.57016583
## [3,] -1.2127405 0.5246863 -0.53661195 -0.04100884
## [4,] 0.8990921 -0.6381722 -0.41782982 -0.68456483
## [5,] -0.4856079 -1.0574103 -1.12197499 -0.21699927
## [6,] -0.2747193 -0.2948713 0.05735485 0.45018384
## X.Genes3.Genes7 X.Genes3.Genes8 X.Genes4.Genes5 X.Genes4.Genes6
## [1,] -1.2289787 0.2062986 0.21256666 -0.82009749
## [2,] 3.2952219 0.5603280 -0.33551056 -0.68389022
## [3,] -0.6615142 -0.8208560 0.07837627 0.77142967
## [4,] -1.1144847 -0.2816493 -0.02322449 -0.41441027
## [5,] -1.2289787 -0.9767993 -1.03239192 0.04376543
## [6,] 0.6031412 0.9669974 -1.03038818 -0.81726379
## X.Genes4.Genes7 X.Genes4.Genes8 X.Genes5.Genes6 X.Genes5.Genes7
## [1,] -0.8343211 1.27464199 -0.62192832 -0.540364461
## [2,] 0.8602911 -0.67289634 -1.16637670 -0.002284382
## [3,] -0.1070488 -0.30912262 -0.59863632 -0.910825826
## [4,] -0.9240035 0.03565863 0.23633828 -0.504340543
## [5,] -1.1552503 -0.84450730 0.01109224 -1.125938435
## [6,] -0.7294656 -0.56757659 -0.64043302 -0.562121158
## X.Genes5.Genes8 X.Genes6.Genes7 X.Genes6.Genes8 X.Genes7.Genes8
## [1,] 1.6829651 1.3868208 -0.12956983 0.11597046
## [2,] -1.0196164 0.2845163 1.23380165 -0.08384636
## [3,] -1.0196164 0.7923786 0.73089992 0.08244763
## [4,] 0.6340071 1.1216686 0.01762371 -0.20021082
## [5,] -0.7815674 0.3919226 -0.30010144 -0.26012786
## [6,] -0.3694684 -0.2546694 -0.17668121 -1.08647434
```

```
Int$interLength
```

```
## X.Genes1.Genes2 X.Genes1.Genes3 X.Genes1.Genes4 X.Genes1.Genes5
## 1 1 1 1
## X.Genes1.Genes6 X.Genes1.Genes7 X.Genes1.Genes8 X.Genes2.Genes3
## 1 1 1 1
## X.Genes2.Genes4 X.Genes2.Genes5 X.Genes2.Genes6 X.Genes2.Genes7
## 1 1 1 1
## X.Genes2.Genes8 X.Genes3.Genes4 X.Genes3.Genes5 X.Genes3.Genes6
## 1 1 1 1
## X.Genes3.Genes7 X.Genes3.Genes8 X.Genes4.Genes5 X.Genes4.Genes6
## 1 1 1 1
## X.Genes4.Genes7 X.Genes4.Genes8 X.Genes5.Genes6 X.Genes5.Genes7
## 1 1 1 1
## X.Genes5.Genes8 X.Genes6.Genes7 X.Genes6.Genes8 X.Genes7.Genes8
## 1 1 1 1
```

Test for potential main or interaction effects is done with the function `GLmodel`. Parameters include `nlambda`, the length of the grid of possible lambda values, `limitLambda` the number of the largest lambda values among

the grid to consider for the cross validation and `lambda.cri` the criteria for lambda selection (minimum or oneSE value).

```
res <- GLmodel(Geno$X, Pheno$y, Int$XBet, interLength=Int$interLength,
  listGenesSNP=Geno$listGenesSNP, nlambda=100, limitLambda=25, lambda.cri="min")
```

```
##
## LinReg()
## Lambda = 1388.49 1100.355 872.0123 691.0549 547.6492 434.0026 343.9396 272.5663 216.0041 171.1796 13
```

The outputs of GLmodel contain:

- the matrix `res_GL.min` giving for each SNP an interaction variable the group lasso coefficient values at the optimal lambda level,
- the matrix `pval.adj` that give adjusted pvalues of each variable with nonzero group lasso coefficient.

```
res
```

```
## $res_GL.min
##              Coefs
## Genes1      0.774115652
## Genes1      1.088838532
## Genes1     -0.212205483
## Genes1     -0.359715714
## Genes1      0.251117271
## Genes1      0.077745993
## Genes2      1.034953657
## Genes2      0.751272211
## Genes2     -0.227326819
## Genes2      0.007677615
## Genes2     -0.186737098
## Genes2      0.397955300
## Genes3      0.097299058
## Genes3      0.161389619
## Genes3      0.016012206
## Genes3     -0.036971423
## Genes3     -0.043448945
## Genes3      0.021385553
## Genes4      0.757663956
## Genes4      0.325003095
## Genes4     -0.139607623
## Genes4     -0.194381959
## Genes4     -0.254329496
## Genes4      0.136103406
## Genes5      0.115314233
## Genes5     -0.136702380
## Genes5      0.158719438
## Genes5     -0.128245444
## Genes5     -0.069903366
## Genes5      0.031734515
## Genes6      0.000000000
## Genes6      0.000000000
## Genes6      0.000000000
```

```

## Genes6      0.000000000
## Genes6      0.000000000
## Genes6      0.000000000
## Genes7      0.000000000
## Genes7      0.000000000
## Genes7      0.000000000
## Genes7      0.000000000
## Genes7      0.000000000
## Genes7      0.000000000
## Genes7      0.000000000
## Genes8      0.000000000
## Genes8      0.000000000
## Genes8      0.000000000
## Genes8      0.000000000
## Genes8      0.000000000
## Genes8      0.000000000
## X.Genes1.Genes2 0.000000000
## X.Genes1.Genes3 0.243086593
## X.Genes1.Genes4 0.000000000
## X.Genes1.Genes5 0.401950190
## X.Genes1.Genes6 0.000000000
## X.Genes1.Genes7 0.000000000
## X.Genes1.Genes8 0.000000000
## X.Genes2.Genes3 0.000000000
## X.Genes2.Genes4 0.000000000
## X.Genes2.Genes5 0.235994954
## X.Genes2.Genes6 0.000000000
## X.Genes2.Genes7 0.000000000
## X.Genes2.Genes8 0.000000000
## X.Genes3.Genes4 1.026008870
## X.Genes3.Genes5 0.000000000
## X.Genes3.Genes6 0.000000000
## X.Genes3.Genes7 0.000000000
## X.Genes3.Genes8 0.000000000
## X.Genes4.Genes5 0.000000000
## X.Genes4.Genes6 0.000000000
## X.Genes4.Genes7 0.140106570
## X.Genes4.Genes8 0.000000000
## X.Genes5.Genes6 0.000000000
## X.Genes5.Genes7 0.000000000
## X.Genes5.Genes8 0.000000000
## X.Genes6.Genes7 0.218978412
## X.Genes6.Genes8 0.000000000
## X.Genes7.Genes8 0.620258051
##
## $pval.adj
##          pval.adj
## Genes1      0.0060
## Genes2      0.0000
## Genes3      0.7680
## Genes4      0.7680
## Genes5      0.7656
## X.Genes1.Genes3 1.0000
## X.Genes1.Genes5 0.9330
## X.Genes2.Genes5 1.0000

```

```

## X.Genes3.Genes4    0.0440
## X.Genes4.Genes7    1.0000
## X.Genes6.Genes7    0.4980
## X.Genes7.Genes8    1.0000
##
## $vc
## $vc$cv.error
##      [,1]      [,2]      [,3]
## [1,] 26.03778 28.20617 30.37455
## [2,] 25.44842 27.56409 29.67975
## [3,] 23.22780 25.11934 27.01087
## [4,] 21.99849 23.75046 25.50242
## [5,] 20.34819 21.95416 23.56013
## [6,] 19.61963 21.17145 22.72326
## [7,] 19.11130 20.62290 22.13450
## [8,] 18.50820 19.97642 21.44464
## [9,] 18.51268 19.99756 21.48243
## [10,] 17.82899 19.28354 20.73808
## [11,] 17.03897 18.46031 19.88165
## [12,] 16.68447 18.12269 19.56090
## [13,] 16.44592 17.82879 19.21167
## [14,] 15.79100 17.14569 18.50038
## [15,] 16.30621 17.66392 19.02162
## [16,] 16.63534 17.98745 19.33957
## [17,] 16.55811 17.92767 19.29724
## [18,] 17.10219 18.50330 19.90440
## [19,] 17.57334 19.03364 20.49394
## [20,] 17.46342 18.82543 20.18744
## [21,] 18.53222 20.07354 21.61485
## [22,] 18.54618 20.01007 21.47396
## [23,] 19.47074 20.97730 22.48385
## [24,] 19.57690 21.05210 22.52729
## [25,] 19.84027 21.46739 23.09450
##
## $vc$lambda
## [1] 1388.490254 1100.354780 872.012346 691.054871 547.649167
## [6] 434.002599 343.939638 272.566281 216.004116 171.179568
## [11] 135.656880 107.505758 85.196474 67.516749 53.505869
## [16] 42.402486 33.603245 26.629997 21.103817 16.724414
## [21] 13.253812 10.503420 8.323780 6.596454 5.227577
##
## $vc$lambda.min
## [1] 67.51675
##
## $vc$lambda.oneSE
## [1] 135.6569
##
## $vc$id.lambda.min
## [1] 14
##
## attr(,"class")
## [1] "GLmodel"

```

The **GGEE** package contains a plot function `plot.GLmodel`. The function takes as parameter a `GLmodel` object and provides a representation of cross validation results. It depicts the value of the cross validation error for each lambda considered and thus allows to identify the optimal lambda values depending of the criteria of interest (minimal or oneSE). This plot allows to verify that enough lambda values was considered for the cross validation. If the curve doesn't show a clear minimal value the parameter `limitLambda` of the `GLmodel` has to be enlarged.

```
plotGLmodel(res)
```

