# The LiquidAssociation Package

Yen-Yi Ho

October 28, 2009

## 1 Introduction

The LiquidAssociation package provides analytical methods to study three-way interactions. It incorporates methods to examine a particular kind of three-way interaction called liquid association (LA). The term liquid association was first proposed by Li (2002). It describes the extend to which the correlation of a pair of variables depends on the value of a third variable. The term "liquid" was used in contrast with "solid" to emphasize that the association between a pair of variables changes according to the value of a third variable. Li first reported the existence of liquid association in the yeast cell cycle gene expression data set by Spellman et al. (1998). Furthermore, the liquid association phenomena could potentially be found in other kind of data as well. Building on the ground-breaking work by Li, Ho et al. (2009) extended liquid association to accommodate more intricate co-dependencencies among the 3 variables, calling the expanded statistic *generalized liquid association*, or GLA for short.

This software package provides functions to implement the estimation of liquid association through two approaches: the direct and the model-based estimation approach. For the model-based approach, we introduce the conditional normal model (CNM) and provided a generalized estimating equations (GEE)-based estimation procedure. In addition, we provide functions to perform hypothesis testing using direct estimate (sGLA) and model-based estimates (GEEb5) in this package.

## 2 Simple Usage

Here we present a typical work flow to investigate liquid association using a gene triplet data. We start with load the R package and the example data. In this package, we use the yeast cell cycle gene expression data by Spellman (1998). The data can be obtained through package *yeastCC*. The annotation package for the yeast experiment *org.Sc.sgd.db* can be obtained through Bioconductor.

```
> library(LiquidAssociation)
> library(yeastCC)
> library(org.Sc.sgd.db)
> data(spYCCES)
> lae <- spYCCES[, -(1:4)]
> lae <- lae[apply(is.na(exprs(lae)), 1, sum) < ncol(lae) * 0.3,
+     ]
> probname <- rownames(exprs(lae))
> genes <- c("HTS1", "ATP1", "CYT1")
> geneMap <- unlist(mget(genes, revmap(org.Sc.sgdGENENAME), ifnotfound = NA))
> whgene <- match(geneMap, probname)
```

After removing genes with high missing percentage, we keep 5721 genes for further analysis. We are interested in three genes: HTS1, ATP1, and CYT1 that are involved in the Yeast electron transport pathway. We would like to know whether the correlation of HTS1 and ATP1 gene can be modulated by the level of CYT1 gene.

```
> data <- t(exprs(lae[whgene, ]))
> eSetdata <- lae[whgene, ]
> data <- data[!is.na(data[, 1]) & !is.na(data[, 2]) & !is.na(data[,
+     3]), ]
> colnames(data) <- genes
> str(data)

 num [1:66, 1:3] 0.3 -0.25 0.18 0.05 0.05 -0.16 0.26 0.14 0.03 0.01 ...
 - attr(*, "dimnames")=List of 2
  ..$ : chr [1:66] "alpha_0" "alpha_7" "alpha_14" "alpha_21" ...
  ..$ : chr [1:3] "HTS1" "ATP1" "CYT1"
```

We notice that after removing missing observations, there are 66 observations left. We can use the *plotGLA* function to examine whether the correlation of the first two genes changes according to the level of the third gene. The *plotGLA* function produces scatter plot conditioning on the level of a third gene, $X_3$. We can specify which column in the data to be the third gene using the argument *dim*. For example, we can specify HTS1 and ATP1 gene as the first two modulated genes, and CYT1 as the third controller gene by setting *dim=3*. The *cut* argument is used to specify the number of grid points over the third controller gene. In the *plotGLA* function, the input *data* can be in ExpressionSet class as well.

```
> plotGLA(data, cut = 3, dim = 3, pch = 16, filen = "GLAplot",
+     save = TRUE)
```

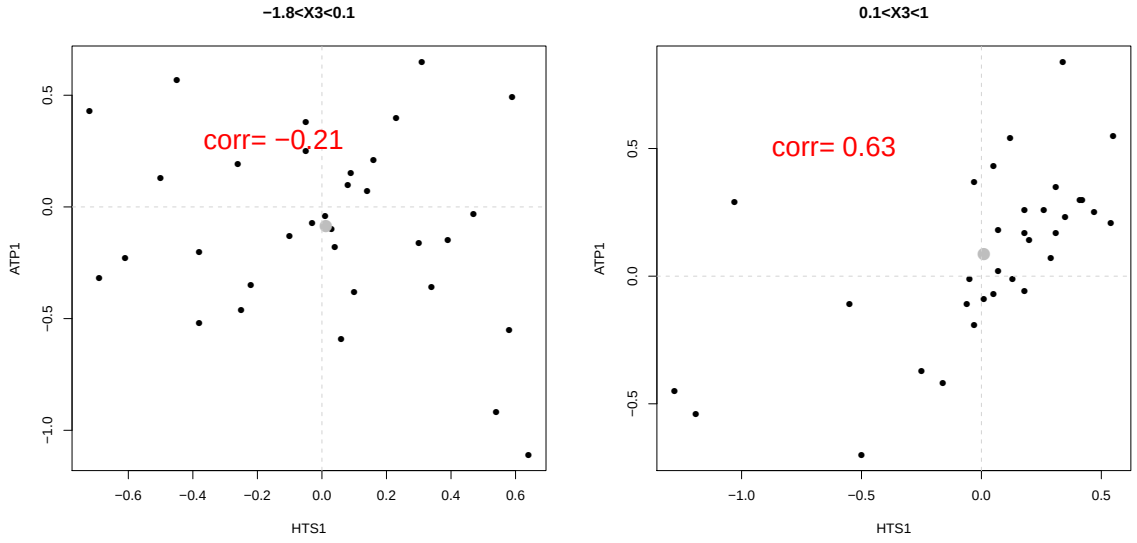


Figure 1: Conditional distributions of (HTS1, ATP1 | CYT1) according to the gene expression level of CYT1.

According to Figure 1, we find the evidence for the existence of liquid association among the triplet since the correlation of HTS1, and ATP1 gene changes from -0.21 to 0.63. We can further perform estimation and hypothesis testing to quantify the strength of liquid association. As described in Li (2002) and Ho et al (2009), the calculation of the liquid association measures assumes all three variable are standardized with mean 0 and variance 1 and the third gene follows normal distribution. Hence in this package, within the *GLA*, *LA*, *CNM.full*, *CNM.simple*, *getsGLA*, *getsLA* functions the standardization and normalization steps are performed internally (so that the three variables are with mean 0 and variance 1, and the third variable is normally distributed through quantile normalization).

We now use GLA static to robustly measure the magnitude of liquid association. The measure ranges from $-\sqrt{2/\pi} \approx -0.798$ to $\sqrt{2/\pi}$. When GLA=0, it means that the correlation of the two modulated genes

does not change according to the level of the third controller gene, hence there is no evidence of LA. In addition, when GLA >0, it indicates that the correlation of the first two genes increases with increasing value of the third gene and vise versa. We use the function *GLA* to calculate the GLA estimate for a given triplet data as follows:

```
> LAest <- LA(data)
> GLAest <- rep(0, 3)
> for (dim in 1:3) {
+     GLAest[dim] <- GLA(data, cut = 4, dim = dim)
+ }
> LAest

LA(HTS1,ATP1,CYT1)
         0.5633369

> GLAest

[1] 0.4162624 0.3085295 0.3219779
```

The data argument in these function can also be in ExpressionSet class as follows:

```
> eSetGLA <- GLA(eSetdata, cut = 4, dim = 3, geneMap = geneMap)
> eSetGLA

GLA(HTS1,ATP1|CYT1)
          0.3219779
```

In the above example, we calculate three GLA estimates by sequentially changing the third controller gene. In addition, the three-product-moment estimator proposed by Li (2002) can also be calculated using the *LA* function as shown above. In the example, we find noticeable differences between the three-prodcut-moment and GLA estimates, LAest and GLAest, respectively. As described in Ho (2009), the GLA estimator is more robust than LA in the sense that it could still correctly capture liquid association even when the marginal mean and variance depend on the third variable.

The second approach to estimate liquid association is using the model-based estimator. We fit the CNM using the function *CNM.full* as shown below. Furthermore, the CNM model is written as:

$$\begin{aligned} X_3 &\sim N(\mu_3, \sigma_3^2) \\ X_1, X_2 | X_3 &\sim N(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \Sigma). \end{aligned}$$

where $\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$. The mean vector $(\mu_1, \mu_2)$ and variance matrix $\Sigma$ depend on the level of $X_3$ as written below:

$$\begin{aligned} \mu_1 &= \beta_1 X_3, \\ \mu_2 &= \beta_2 X_3, \\ \log \sigma_1^2 &= \alpha_3 + \beta_3 X_3, \\ \log \sigma_2^2 &= \alpha_4 + \beta_4 X_3, \\ \log [\frac{1+\rho}{1-\rho}] &= \alpha_5 + \beta_5 X_3. \end{aligned}$$

```
> FitCNM.full <- CNM.full(data)
> FitCNM.full
```

```
Model: CNM(HTS1,ATP1|CYT1)
     estimates     san.se         wald      p value
a3 -0.01231179 0.21643955  0.003235709 9.546382e-01
a4 -0.27844712 0.17085119  2.656130850 1.031514e-01
a5  0.91196967 0.44923061  4.121184951 4.234941e-02
b1  0.02594554 0.06768601  0.146935788 7.014810e-01
b2  0.41278567 0.08571134 23.193840594 1.464657e-06
b3  0.03436928 0.22654459  0.023016197 8.794150e-01
b4 -0.18970033 0.20901616  0.823714018 3.640965e-01
b5  1.36133282 0.51779777  6.912072351 8.561570e-03
```

The main parameter of interest in the CNM for examining the existence of liquid association is $b_5$. As shown in the result above, the GEEb5 Wald test statistic is displayed in the 8th row 3rd column (GEEb5=6.91) with p value 0.009. We conclude there is statistically significant evidence that indicates the existence of liquid association for (HTS1, ATP1 | CYT1). We can also perform hypothesis testing using the direct estimate approach using the *getsGLA* function as follows:

```
> sGLAest <- getsGLA(data, boots = 20, perm = 50, cut = 4, dim = 3)
> sGLAest

    sGLA  p value
2.750812 0.000000

> sLAest <- getsLA(data, boots = 20, perm = 50)
> sLAest

     sLA  p value
5.186875 0.000000
```

where the argument *boots* specify the number of bootstrap iteration for estimating the bootstrap standard error. For demonstration purpose, we set boots to 20. The *perm* argument specifies the number of iterations to calculate permuted p value. In addition, we can also perform hypothesis testing using sLA statistic based on the three-product-moment estimator proposed by Li (2002). In real applications, sGLA is likely to be more robust than sLA. We draw similar conclusion using sGLA test statistic and GEEb5 with p value less than 0.01.

# 3 Extended Examples

In this section, we demonstrate an typical analysis procedure using the entire gene expression data set. In this example, we first filter out genes with small variances and keep 150 genes for further analysis.

```
> lae <- t(exprs(lae))
> V <- apply(lae, 2, var, na.rm = TRUE)
> ibig <- V > 0.5
> sum(ibig)

[1] 150

> big <- which(ibig)
> bigtriplet <- lae[, big]
> dim(bigtriplet)

[1]  73 150
```

We can annotation the ORF ID in the data set to their gene names as follows:

```
> x <- org.Sc.sgdGENENAME
> mappedgenes <- mappedkeys(x)
> xx <- as.list(x[mappedgenes])
> mapid <- names(xx)
> orfid <- colnames(bigtriplet)
> genename1 <- xx[match(orfid, mapid)]
> imap <- which(sapply(genename1, length) != 1)
> bigtriplet <- bigtriplet[, -imap]
> colnames(bigtriplet) <- genename1[-imap]
```

After removing ORF probe set that can not be mapped to genes, a total number of 383306 possible triplet combinations that can be generated by the 133 genes. We now calculate the GLA estimates for all possible triplet combinations. Here, we demonstrate calculating GLA estimates for triplet combination #1 to #100.

```
> num <- choose(ncol(bigtriplet), 3)
> pick <- t(combn(1:ncol(bigtriplet), 3))
> GLAout <- matrix(0, nrow = 100, 3)
> for (i in 1:100) {
+     dat1 <- bigtriplet[, pick[i, ]]
+     for (dim in 1:3) {
+         GLAout[i, dim] <- GLA(dat1, cut = 4, dim = dim)
+     }
+ }
```

We would like to find triplets with large GLA estimates. For example, we can choose triplet with GLA greater than 0.2 as follows:

```
> GLAmax <- apply(abs(GLAout), 1, max)
> imax <- which(GLAmax > 0.2)
> pickmax <- pick[imax, ]
> GLAmax <- GLAout[imax, ]
> trip.order <- t(apply(t(abs(GLAmax)), 2, order, decreasing = TRUE))
```

Furthermore, we perform hypothesis testing using the first triplet as demonstration. Based on the analysis result, we can not reject the null hypothesis that the liquid association is 0. We draw the same conclusion using GEEb5 and sGLA statistics.

```
> whtrip <- 1
> data <- bigtriplet[, pickmax[whtrip, ]]
> data <- data[, trip.order[whtrip, ]]
> data <- data[!is.na(data[, 1]) & !is.na(data[, 2]) & !is.na(data[,
+     3]), ]
> data <- apply(data, 2, qqnorm2)
> data <- apply(data, 2, stand)
> FitCNM1 <- CNM.full(data)
> FitCNM1

Model: CNM(SEO1,RFA1|SSA1)
     estimates    san.se       wald     p value
a3 -0.02131184 0.1577184 0.01825901 0.892512334
a4 -0.13771132 0.1694611 0.66038843 0.416422847
a5 -0.32762519 0.2696887 1.47580482 0.224432053
b1  0.06840792 0.1159649 0.34798432 0.555256383
b2 -0.05317252 0.1093997 0.23623402 0.626939435
b3 -0.04523443 0.1461961 0.09573412 0.757010192
b4 -0.49465917 0.1786326 7.66814787 0.005620412
b5 -0.71723661 0.4098913 3.06187546 0.080149111
```

```
> sGLA1 <- getsGLA(data, boots = 20, perm = 50, cut = 4, dim = 3)
> sGLA1

     sGLA   p value
-1.277645  0.360000
```

# 4 Reference

# References


Ho, Y.-Y., Cope, L. M., Louis, T. A., and Parmigiani, G. (2009). Generalized liquid association. Technical report, Johns Hopkins University, Dept. of Biostatistics Working Papers. Working Paper 183.

Li, K.-C. (2002). Genome-wide coexpression dynamics: theory and application. *Proc Natl Acad Sci U S A* **99,** 16875–16880.

Spellman, P. T., Sherlock, G., Zhang, M. Q., Iyer, V. R., Anders, K., Eisen, M. B., Brown, P. O., Botstein, D., and Futcher, B. (1998). Comprehensive identification of cell cycle-regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Mol Biol Cell* **9,** 3273–3297.


# 5 Session Information

- R version 2.8.1 (2008-12-22), `x86_64-unknown-linux-gnu`
- Locale: `LC_CTYPE=en_US.iso885915;LC_NUMERIC=C;LC_TIME=en_US.iso885915;LC_COLLATE=en_US.iso885915;LC_`
- Base packages: base, datasets, graphics, grDevices, methods, stats, tools, utils
- Other packages: AnnotationDbi 1.4.2, Biobase 2.2.2, DBI 0.2-4, geepack 1.0-13, LiquidAssociation 1.0.4, org.Sc.sgd.db 2.2.6, RSQLite 0.7-1, yeastCC 1.2.6