# Integration of muliple tables

in

Methods to integrate multiple tables in biomedical studies to detect biomarkers and stratify individuals

Instituto de Salud Carlos III. Centro Nacional de Epidemiología
September, 2017

Juan R Gonzalez

BRGE - Bioinformatics Research Group in Epidemiology
Barcelona Institute for Global Health (ISGlobal)
e-mail:juanr.gonzalez@isglobal.org
http://www.creal.cat/brge
and Departament of Mathematics, UAB

# Outline


1. Introduction
2. Integrating two or more omic datasets
3. RGCCA
4. SGCCA
5. Recommended lectures

# Outline

1 Introduction
2 Integrating two or more omic datasets
3 RGCCA
4 SGCCA
5 Recommended lectures

# Introduction

- Undersanding the genetic basis of complex traits is an open question for may researchers
- Recent advances in technology has made this problem even more complex by incorporating other pieces of information such as neuroimaging, daily measurements of enviromental exposures among others
- The main challenge is to analyze the vast amount of data that is being generated, particularly how to integrate information from different tables

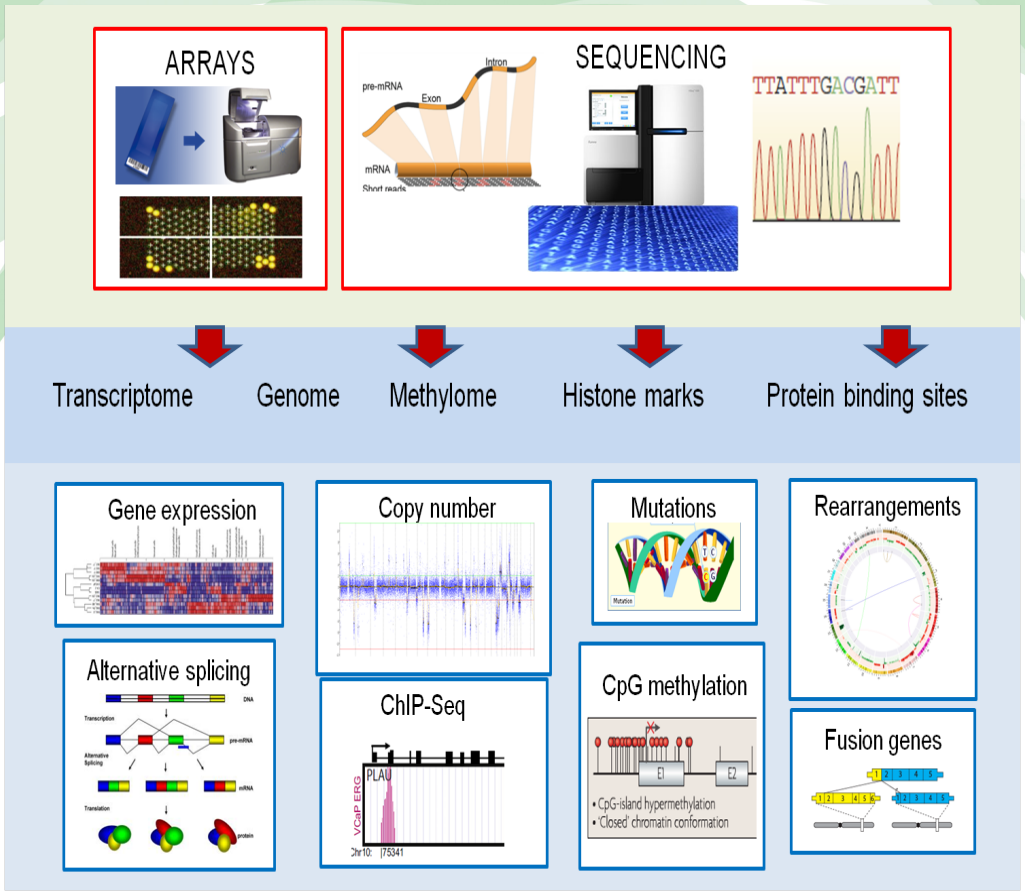
# Introduction

- Undersanding the genetic basis of complex traits is an open question for may researchers
- Recent advances in technology has made this problem even more complex by incorporating other pieces of information such as neuroimaging, daily measurements of enviromental exposures among others
- The main challenge is to analyze the vast amount of data that is being generated, particularly how to integrate information from different tables

# Introduction

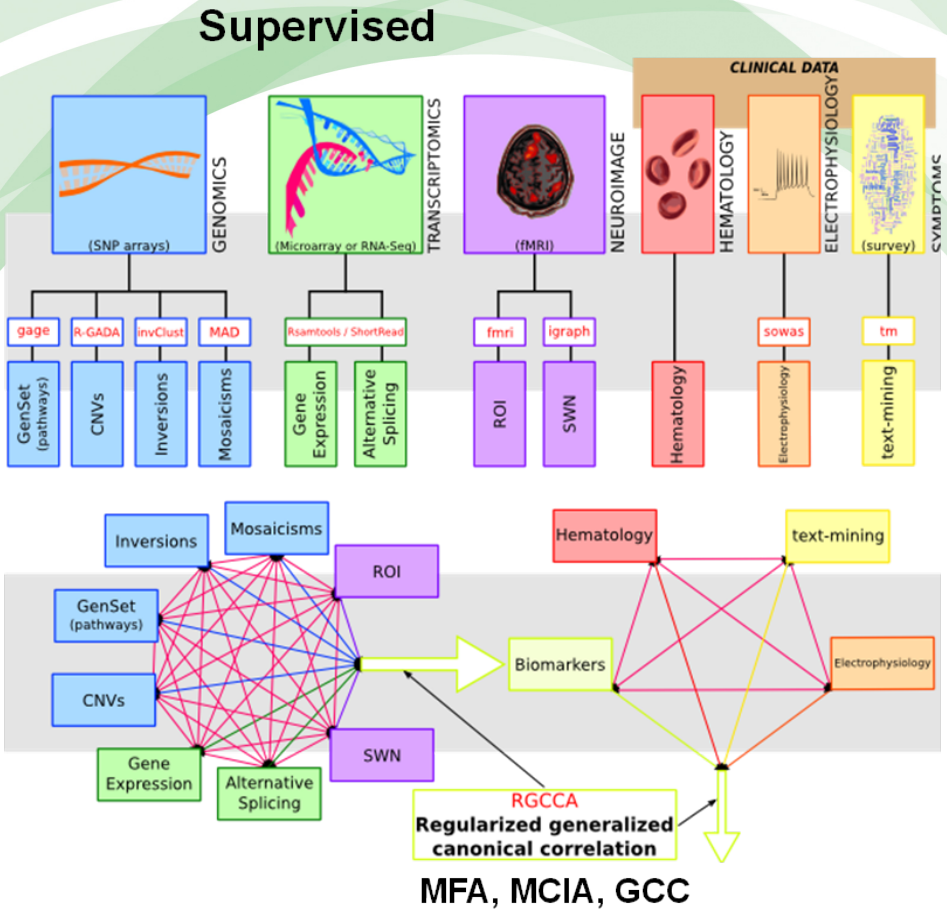
- Undersanding the genetic basis of complex traits is an open question for may researchers
- Recent advances in technology has made this problem even more complex by incorporating other pieces of information such as neuroimaging, daily measurements of enviromental exposures among others
- The main challenge is to analyze the vast amount of data that is being generated, particularly how to integrate information from different tables

# Introduction

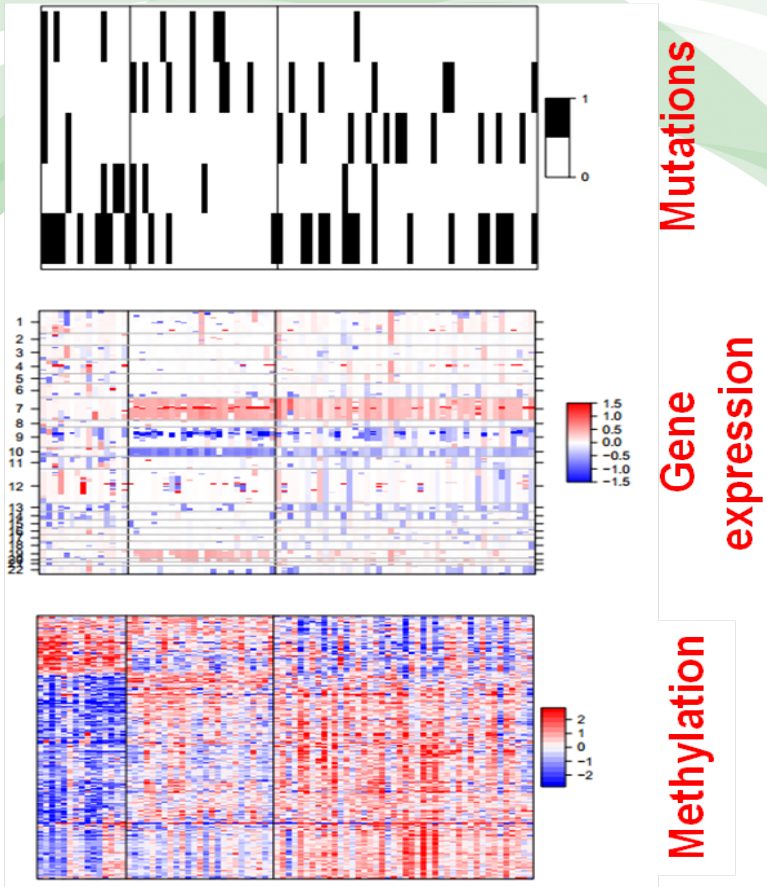



100s-1000s of aberrations

# Introduction



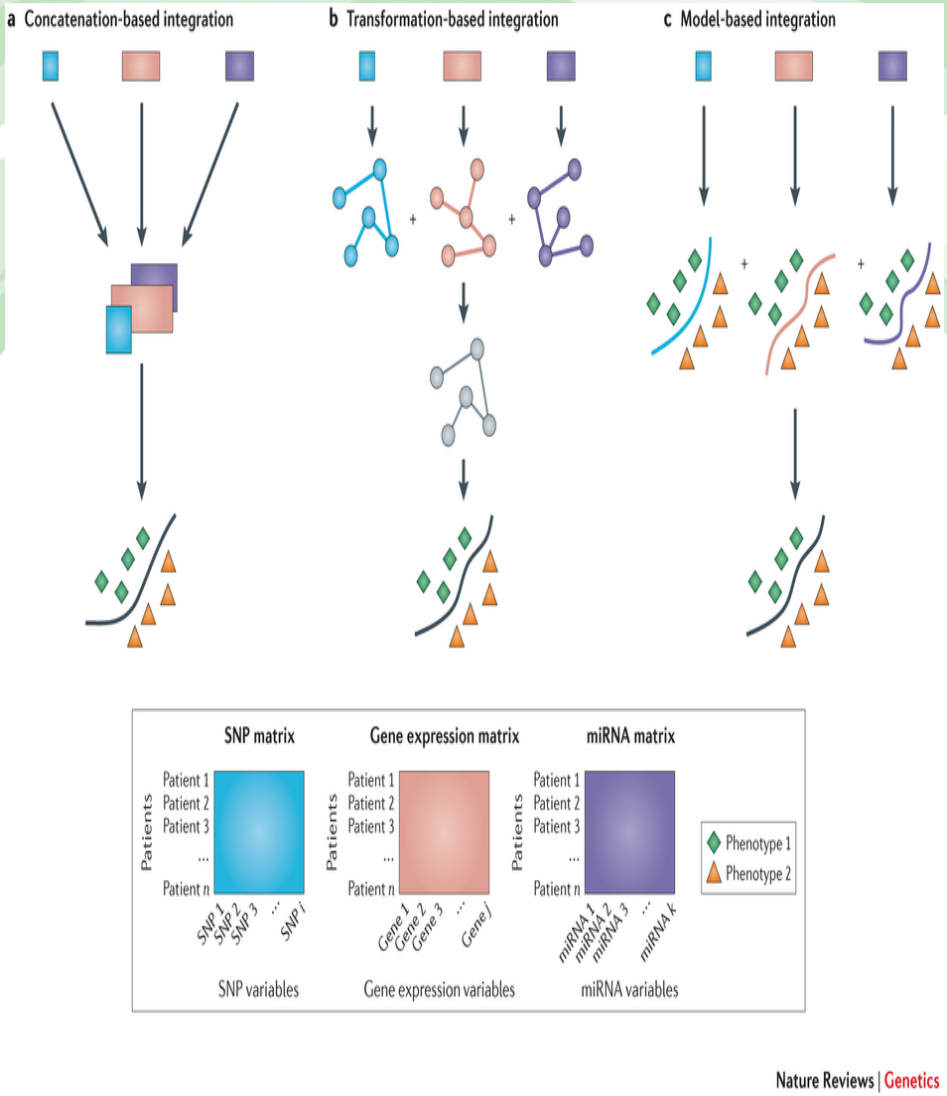
# Introduction



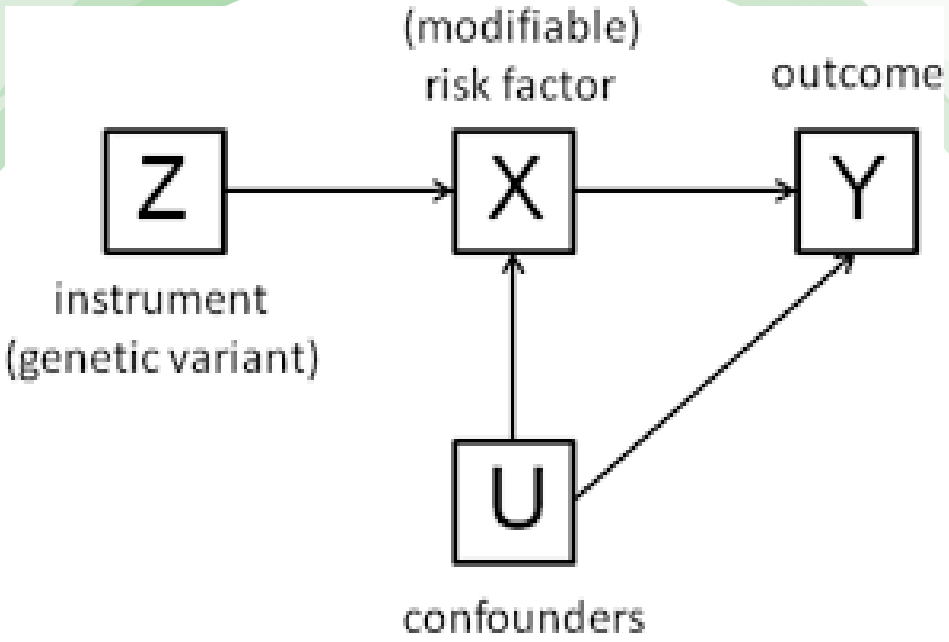
# Introduction

- Data integration (Integrative bioinformatics, integrated analysis, crossomics, multi-dataset analysis, data fusion, ...) is being crucial in Bioinformatics/Biology/Epidemiology
- Data integration may refer to different aspects
  - Computational combination of data (sets)
  - Simultaneous analysis of different variables from different tables, different time points, different tissues, ...
  - Provide biological insights by using information from existing databases (ENCODE, GTEX, KEGG, ...)
- Here we mean the process by which different types of data are combined as predictor variables to allow more thorough and comprehensive modelling of complex traits or phenotypes

# Types of meta-dimensional analyses

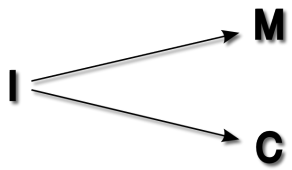


# Mendelian randomization



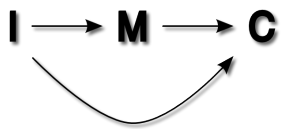
# Conditional Tests

A) Independencia / Asociación

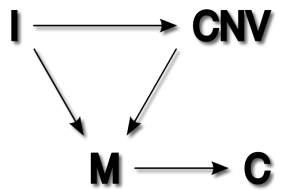



B) Causal


C) Causal / Asociación




D) Complejo



# Outline

1 Introduction

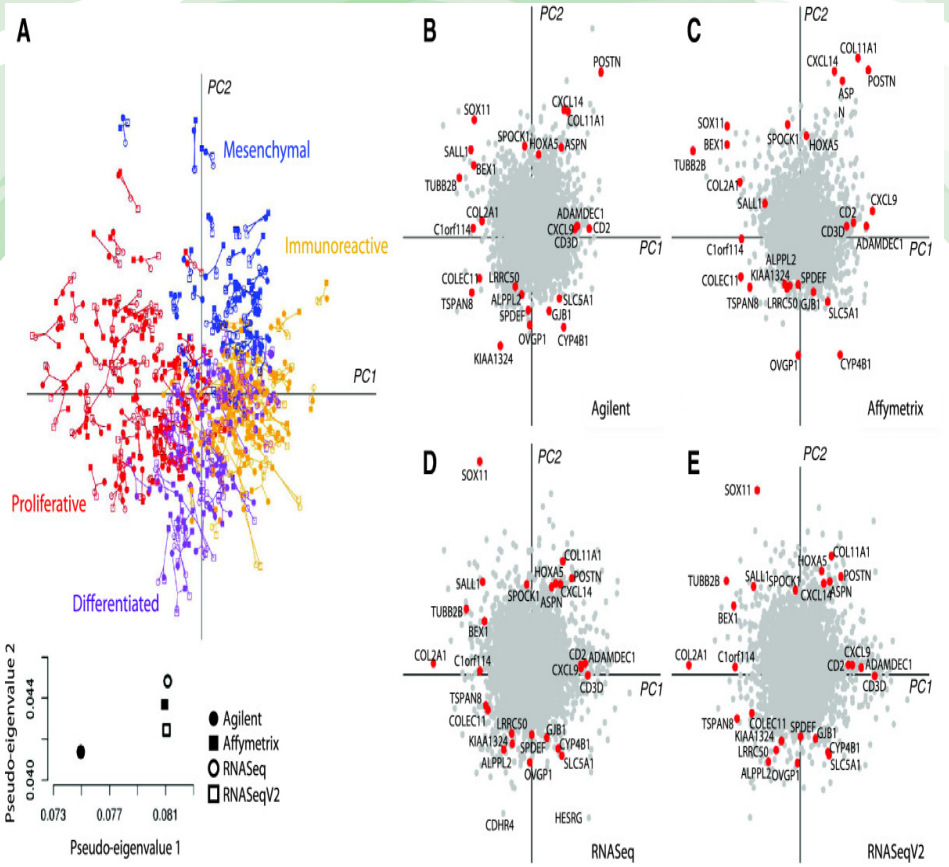
2 Integrating two or more omic datasets

3 RGCCA

4 SGCCA

5 Recommended lectures

# Multiple dataset: dimensionality reduction



# Integrating two or more datasets

- There are methods based on dimension reduction techniques: generalized SVD, Co-Inertia Analysis (CIA), sparse or penalized extensions of Partial Least Squares (PLS), Canonical Correlation (CCA), Multiple Factor Analysis (MFA), Generalized Canonical Correlation (GCA)
- There are methods that are based on path modelling: structural equation models (SEM)
- Reguralized Generalized Canonical Correlation (RGCA) provides a unified framework for different approaches (Tenenhaus, M. and Tenenhaus, A. Psychometrika, 2011, 76(2):257-284). Other methods are special cases of RGCA.
- RGCA integrates a feature selection method, named sparse GCCA (SGCCA) (Tenenhaus, A et al. Biostatistics, 2014, 15(3):569-83).

# Integrating two or more datasets

- There are methods based on dimension reduction techniques: generalized SVD, Co-Inertia Analysis (CIA), sparse or penalized extensions of Partial Least Squares (PLS), Canonical Correlation (CCA), Multiple Factor Analysis (MFA), Generalized Canonical Correlation (GCA)
- There are methods that are based on path modelling: structural equation models (SEM)
- Reguralized Generalized Canonical Correlation (RGCA) provides a unified framework for different approaches (Tenenhaus, M. and Tenenhaus, A. Psychometrika, 2011, 76(2):257-284). Other methods are special cases of RGCA.
- RGCA integrates a feature selection method, named sparse GCCA (SGCCA) (Tenenhaus, A et al. Biostatistics, 2014, 15(3):569-83).

# Integrating two or more datasets

- There are methods based on dimension reduction techniques: generalized SVD, Co-Inertia Analysis (CIA), sparse or penalized extensions of Partial Least Squares (PLS), Canonical Correlation (CCA), Multiple Factor Analysis (MFA), Generalized Canonical Correlation (GCA)
- There are methods that are based on path modelling: structural equation models (SEM)
- Reguralized Generalized Canonical Correlation (RGCA) provides a unified framework for different approaches (Tenenhaus, M. and Tenenhaus, A. Psychometrika, 2011, 76(2):257-284). Other methods are special cases of RGCA.
- RGCA integrates a feature selection method, named sparse GCCA (SGCCA) (Tenenhaus, A et al. Biostatistics, 2014, 15(3):569-83).

# Integrating two or more datasets

- There are methods based on dimension reduction techniques: generalized SVD, Co-Inertia Analysis (CIA), sparse or penalized extensions of Partial Least Squares (PLS), Canonical Correlation (CCA), Multiple Factor Analysis (MFA), Generalized Canonical Correlation (GCA)
- There are methods that are based on path modelling: structural equation models (SEM)
- Reguralized Generalized Canonical Correlation (RGCA) provides a unified framework for different approaches (Tenenhaus, M. and Tenenhaus, A. Psychometrika, 2011, 76(2):257-284). Other methods are special cases of RGCA.
- RGCA integrates a feature selection method, named sparse GCCA (SGCCA) (Tenenhaus, A et al. Biostatistics, 2014, 15(3):569-83).

# Integrating two or more datasets

- Canonical Correlation can be seen as an extension of PCA for more than two tables $X$ and $Y$
- The two datasets can be decomposed as:

$$f = Xp$$

$$g = Yq$$

  where $p$ and $q$ are the loading vectors
- CCA searches for association or correlations among $X$ and $Y$ by

$$\arg\max_{p^i q^i} \text{cor}(Xp^i Yq^i)$$

  for the $i$-th component
- $Xp^i$ and $Yq^i$ are known as canonical variates and their correlations are the canonical correlations.

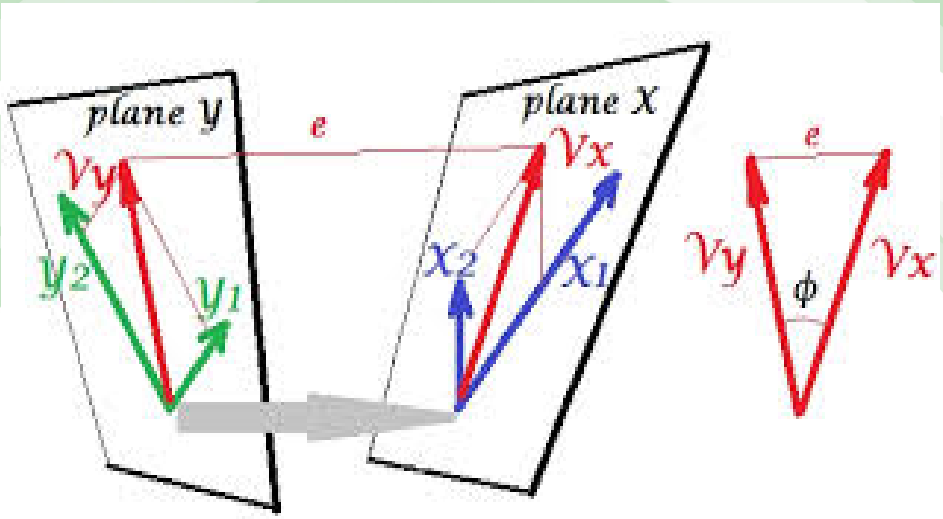
# Integrating two or more datasets

- ($p >> n$) is an issue. Additionally, there is often presence of multicollinearity within both sets of variables that requires a reguralization step.
- This may accomplished by addign a ridge penalty, that is, adding a multiple of the identity matrix to the correlation/covariance matrix.

$$\arg\max_{p^i q^i} \text{cor}(Xp^i Yq^i) + \lambda \mathbf{I}$$

- A sparse solution (filtering the number of variables) is the solution: pCCA, sCCA, CCA-l1, CCA-EN, CCA-group sparse have been used to integrate two omic data.

# Integrating two or more datasets




$V_y$ and $V_x$ are selected to maximize:

- Correlation (CCA)
- Squared Covariance (CIA)

$$\arg\max_{p^i q^i} \mathrm{cov}^2(Xp^i\, Yq^i)$$

# Canonical Correlation

- CCA has been used in omic data
- The main limitation is that the number of features generally greatly exceeds the number of observations
- Consequence 1: parameter estimation cannot be applied using standard methods
- Consequence 2: Most of markers are having no effect in the canonical axes (e.g. almost all components in a and b are 0)
- Penalized (sparse) CCA has been proposed
- Main advantages: More than two tables, easy interpretation (do not need computing p-values nor correcting for multiple comparisons)

# Sparse canonical correlation

$$
\begin{aligned}
U_1 &= a_{11}X_1 + a_{12}X_2 + \cdots + a_{1p}X_p \\
U_2 &= a_{21}X_1 + a_{22}X_2 + \cdots + a_{2p}X_p \\
&\vdots \\
U_p &= a_{p1}X_1 + a_{p2}X_2 + \cdots + a_{pp}X_p \\
\\
V_1 &= b_{11}Y_1 + b_{12}Y_2 + \cdots + b_{1q}Y_q \\
V_2 &= b_{21}Y_1 + b_{22}Y_2 + \cdots + b_{2q}Y_q \\
&\vdots \\
V_p &= b_{p1}Y_1 + b_{p2}Y_2 + \cdots + b_{pq}Y_q
\end{aligned}
$$

a and b for i-th cannonical variable is obtained by maximizing

$$\rho_i^* = \frac{\operatorname{cov}(U_i, V_i)}{\sqrt{\operatorname{var}(U_i)\operatorname{var}(V_i)}}$$

Subject to:

$\|a\|^2 \leq 1$ and $\|b\|^2 \leq 1$

# Co-Inertia

- CIA does not requires an inversion step of the covariance matrix; thus, regularization or penalization implementation is not required
- CIA can deal with disperse variables
- CIA does consider quantitative or qualitative variables
- Can weight cases
- The method provides the RV coefficient. This is a measure of global similarity between the datasets, and is a number between 0 and 1. The closer it is to 1 the greater the global similarity between the two datasets.

# Data analysis illustration

- Data from the Cancer Genome Atlas (TCGA) will be analyzed.
- A subset of the TCGA breast cancer study from Nature 2012 publication have been selected.
- Data `https://tcga-data.nci.nih.gov/docs/publications/brca_`
- Available data are: miRNA, miRNAprecursor, RNAseq, Methylation, proteins from a RPPA array, and GISTIC SNP calls (CNA and LOH). Clinical data are also available.
- We are interested in comparing women with ER+ vs ER-.

```
load("data/breast_TCGA.RData")
group <- droplevels(breast_multi$clin$ER.Status)
```

# Canonical Correlation: gene expression and proteins

```
require(CCA)
df1 <- t(breast_multi$RNAseq)[,1:1000]
df2 <- t(breast_multi$RPPA)
```

```
resCC <- cc(df1, df2)
```

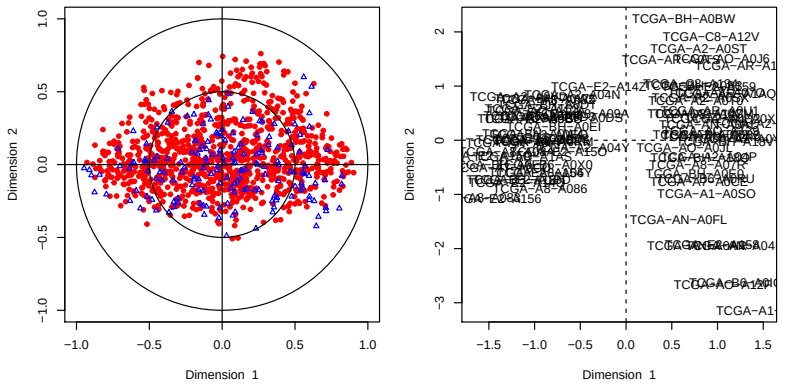
```
Error en chol.default(Bmat) :
  la submatriz de orden 81 no es definida positiva
```

```
resRCC <- rcc(df1, df2, 0.2, 0.1)
```

```
regul <- estim.regul(df1, df2)
resRCC2 <- rcc(df1, df2, regul$lambda1, regul$lambda2)
```

```
plt.cc(resRCC)
```

# Canonical Correlation: gene expression and proteins



# Canonical Correlation: gene expression and proteins

```
require(PMA)
ddlist <- list(df1, df2)
perm.out <- MultiCCA.permute(ddlist,
                             type=c("standard", "standard"),
                             trace=FALSE)
```

```
resMultiCCA <- MultiCCA(ddlist,
                        penalty=perm.out$bestpenalties,
                        ws=perm.out$ws.init,
                        type=c("standard", "standard"),
                        ncomponents=1, trace=FALSE, standardize=TRUE)
```

NOTE: setting `type` equal to "ordered" allows to consider that features are correlated (e.g. genomic regions)

# Canonical Correlation: gene expression and proteins

```
rownames(resMultiCCA$ws[[1]]) <- colnames(df1)
rownames(resMultiCCA$ws[[2]]) <- colnames(df2)
head(resMultiCCA$ws[[1]])
```

```
##                  [,1]
## CREB3L1   0.030502191
## PNMA1     0.035339312
## MMP2      0.000000000
## C10orf90 -0.009616231
## GPR98     0.049652420
## APBB2     0.084521773
```

```
head(resMultiCCA$ws[[2]])
```

```
##                    [,1]
## c.Myc       -0.03022533
## HER3         0.00000000
## XBP1         0.00000000
## Fibronectin  0.00000000
## PAI.1        0.00000000
## p21          0.00000000
```

NOTE: we are interested in selecting those features having a coefficient (`ws`) different from 0.

# Co-inertia: gene expression and proteins

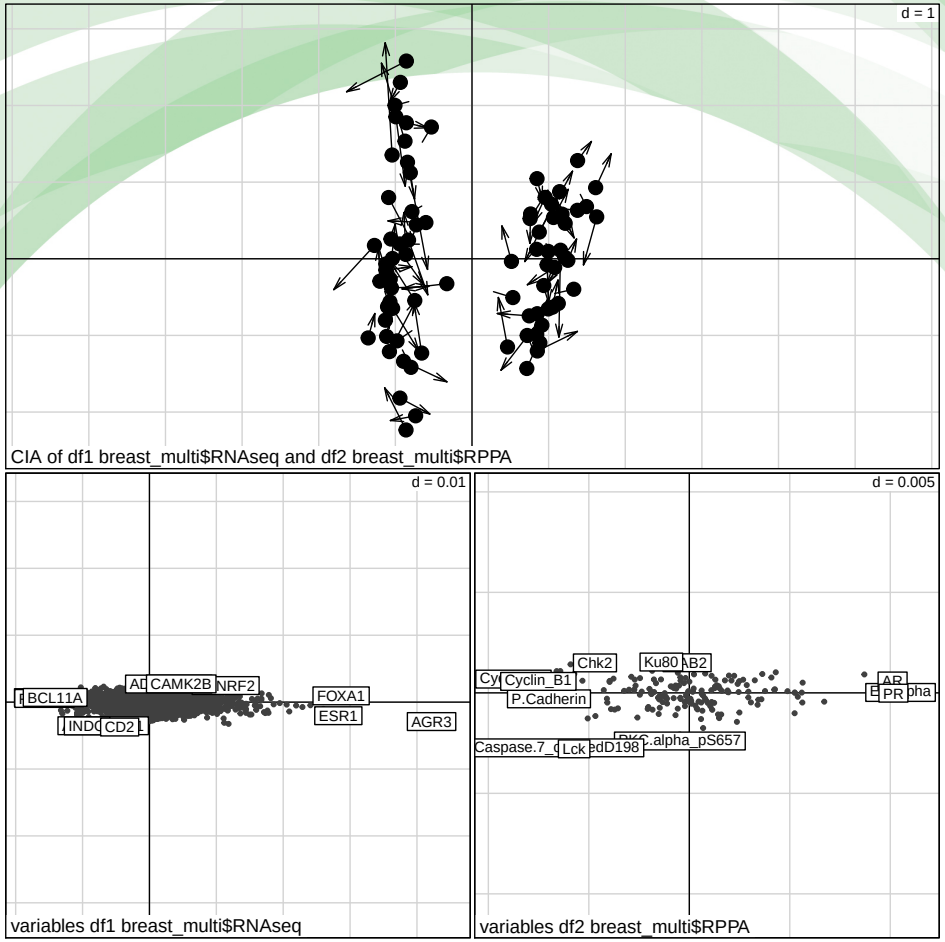
Let us load required packages (coinertia and multiple coinertia)

```
library(made4)
library(omicade4)
```

```
resCIA <- cia(breast_multi$RNAseq, breast_multi$RPPA)
```

```
plot(resCIA, classvec=group, nlab=3, clab=0, cpoint=3 )
```

# Co-inertia: gene expression and proteins



# Co-inertia: gene expression and proteins

Top-5 features of the first axis (positive side) can be retrieve by

```
topVar(resCIA, axis=1, topN=5, end="positive")
```

```
##   ax1_df1_positive ax1_df2_positive
## 1             AGR3         ER.alpha
## 2            FOXA1               PR
## 3             ESR1               AR
## 4             AGR2           INPP4B
## 5         C1orf64            GATA3
```

# Co-inertia: gene expression and proteins

Top-5 features of the first axis (negative side) can be retrieve by

```
topVar(resCIA, axis=1, topN=5, end="negative")
```

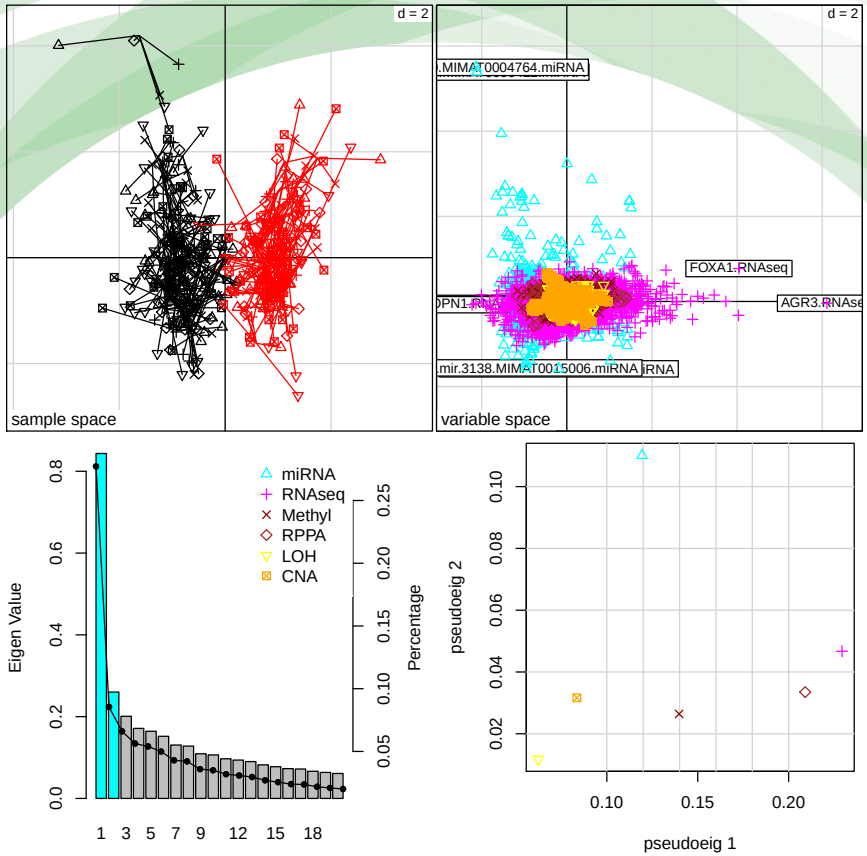
```
##   ax1_df1_negative      ax1_df2_negative
## 1            ROPN1             Cyclin_E1
## 2          ROPN1B             Cyclin_B1
## 3          BCL11A            P.Cadherin
## 4            ART3                  MSH6
## 5           SFRP1 Caspase.7_cleavedD198
```

# More than two tables

```
resMCIA <- mcia( breast_multi[ c(1,3,4,5,6,7) ] )
```

```
plot(resMCIA, axes=1:2, sample.lab=FALSE, sample.legend=FALSE,
     phenovec=group, gene.nlab=2,
     df.color=c("cyan", "magenta", "red4", "brown","yellow", "orange")
     df.pch=2:7)
```

# More than two tables



# More than two tables

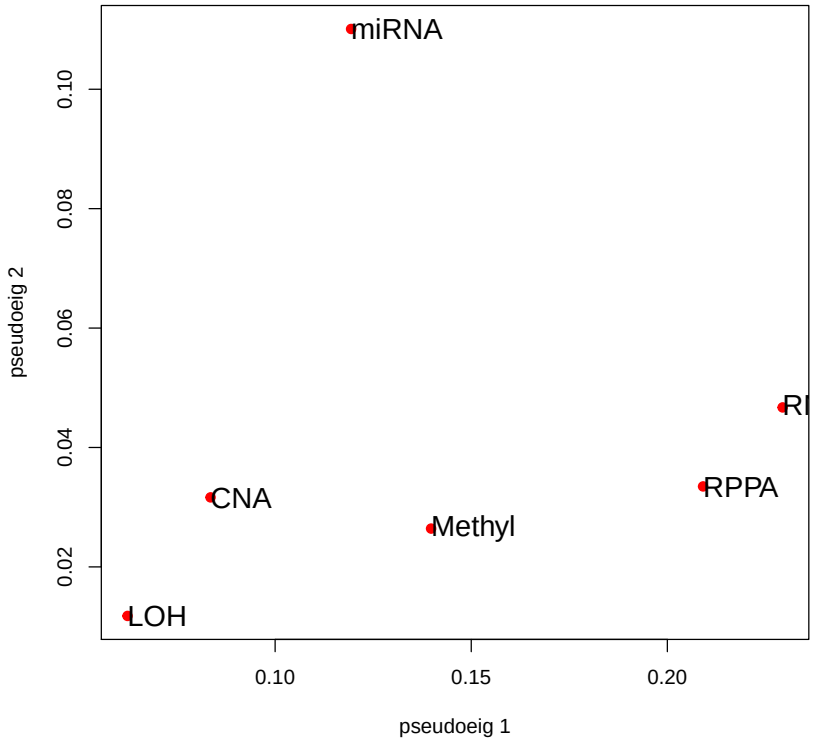
Top-5 features of the first axis (positive side) can be retrieve by

```
topVar(resMCIA, end="positive", axis=1, topN=5)
```

```
##                 ax1_miRNA_positive ax1_RNAseq_positive ax1_Methyl_po
## 1 hsa.mir.4254.MIMAT0016884.miRNA          AGR3.RNAseq   cg08097882.
## 2 hsa.mir.3945.MIMAT0018361.miRNA         FOXA1.RNAseq   cg09952204.
## 3 hsa.mir.302b.MIMAT0000715.miRNA          ESR1.RNAseq   cg04988423.
## 4 hsa.mir.1265.MIMAT0005918.miRNA          AGR2.RNAseq   cg00679738.
## 5 hsa.mir.3171.MIMAT0015046.miRNA      C1orf64.RNAseq   cg12601757.
##   ax1_RPPA_positive ax1_LOH_positive ax1_CNA_positive
## 1     ER.alpha.RPPA        X4006.LOH        X8374.CNA
## 2           AR.RPPA        X4007.LOH        X8381.CNA
## 3           PR.RPPA        X4008.LOH        X8382.CNA
## 4       INPP4B.RPPA        X4009.LOH        X8383.CNA
## 5        GATA3.RPPA        X4010.LOH        X8384.CNA
```

# More than two tables

```
plot(resMCIA$mcoa$cov2,  xlab = "pseudoeig 1",
     ylab = "pseudoeig 2", pch=19, col="red")
text(resMCIA$mcoa$cov2, labels=rownames(resMCIA$mcoa$cov2),
     cex=1.4, adj=0)
```



# Outline

1 Introduction

2 Integrating two or more omic datasets

3 RGCCA

4 SGCCA

5 Recommended lectures

# RGCCA

- RGCCA allows to combine tables with different types of variables between blocks
- The objective of RGCCA is to find, for each block (talbe), a weighted composite of variables (called block component) $\mathbf{y}_j = \mathbf{X}_j\mathbf{a}_j, j = 1, \ldots, J$ (where $\mathbf{a}_j$ is a column-vector with $p_j$ elements) summarizing the relevant information between and within the blocks.
- The block components are obtained such that (i) block components explain well their own block and/or (ii) block components that are assumed to be connected are highly correlated.
- RGCCA has been extended to integrate a variable selection procedure, called SGCCA, allowing the identification of the most relevant features.

# RGCCA

RGCCA (Tenenhaus, M. et al. Psychometrika, 2001 and Tenenhaus A et al, Psychometrika, 2017) is defined as the next optimization problem:

$$\underset{\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_J}{\text{maximize}} \sum_{j,k=1}^{J} c_{jk} g(\text{cov}(\mathbf{X}_j\mathbf{a}_j, \mathbf{X}_k\mathbf{a}_k))$$

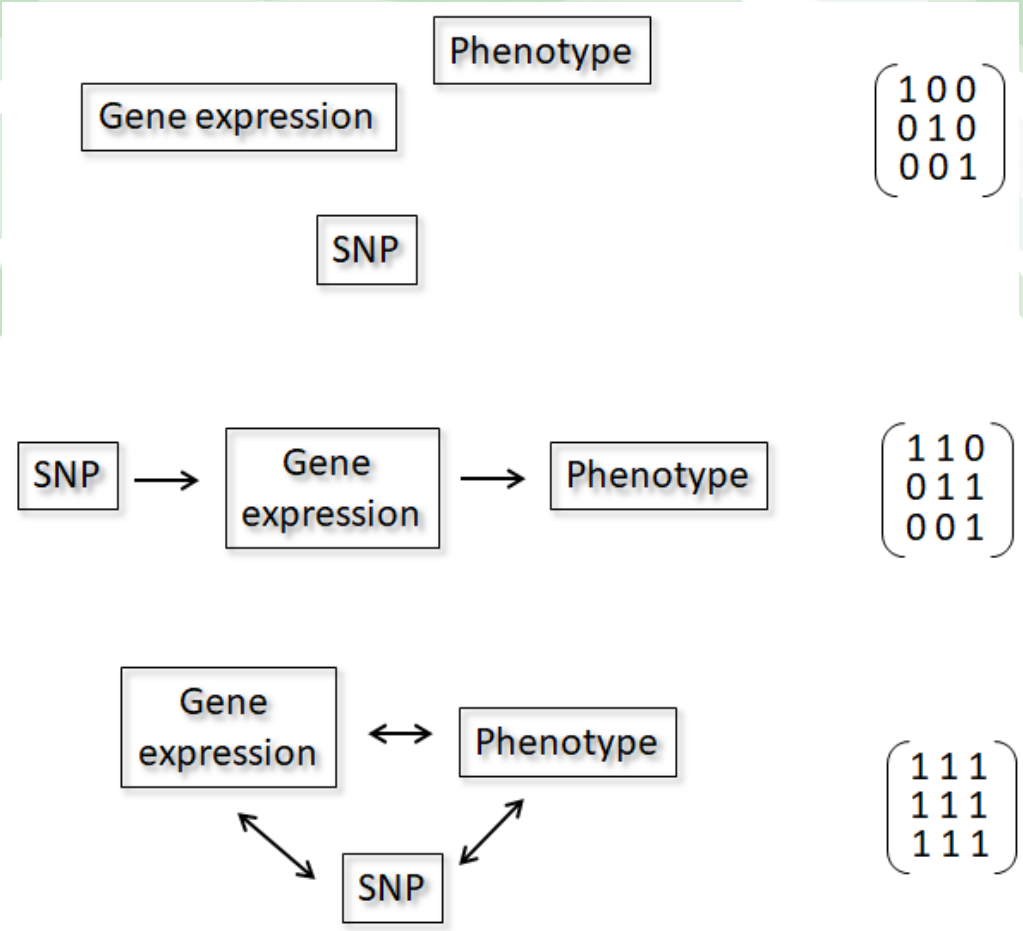
$$\text{s.t.} \quad (1\text{-}\tau_j)\text{var}(\mathbf{X}_j\mathbf{a}_j) + \tau_j\|\mathbf{a}_j\|^2 = 1, j = 1, \ldots, J$$

# RGCCA

- The scheme function $g$ is any continuous convex function and allows to consider different optimization criteria.
  - identity (horst scheme, leading to maximizing the sum of covariances between block components)
  - the absolute value (centroid scheme, yielding maximization of the sum of the absolute values of the covariances)
  - the square function (factorial scheme, thereby maximizing the sum of squared covariances)
  - any even integer $m$, $g(x) = x^m$
- A fair model is a model where all blocks contribute equally to the solution ($m = 1$)
- $m > 1$ is preferable if the user wants to discriminate between blocks.
- In practice, $m$ is equal to 1, 2 or 4. The higher the value of $m$ the more the method acts as block selector

The design matrix **C** is a symmetric $J \times J$ matrix of nonnegative elements describing the network of connections between blocks that the user wants to take into account. Usually, $c_{jk} = 1$ for two connected blocks and 0 otherwise.

# RGCCA


Phenotype
Gene expression
SNP
1 0 0
0 1 0
0 0 1
SNP
Gene expression
Phenotype
1 1 0
0 1 1
0 0 1
Gene expression
Phenotype
SNP
1 1 1
1 1 1
1 1 1

# RGCCA

The $\tau_j$ are called shrinkage parameters ranging from 0 to 1 and interpolate smoothly between maximizing the covariance and maximizing the correlation

- $\tau_j = 1$ yields the maximization of a covariance-based criterion. It is recommended when the user wants a stable component (large variance) while simultaneously taking into account the correlations between blocks. The user must, however, be aware that variance dominates over correlation.
- $\tau_j = 0$ yields the maximization of a correlation-based criterion. It is recommended when the user wants to maximize correlations between connected components. This option can yield unstable solutions in case of multi-collinearity and cannot be used when a data block is rank deficient (e.g. $n < p_j$).
- $0 < \tau_j < 1$ is a good compromise between variance and correlation: the block components are simultaneously stable and as well correlated as possible with their connected block components. This setting can be used when the data block is rank deficient.

# PCA as RGCCA

Principal Component Analysis is defined as the following optimization problem

$$\underset{\mathbf{a}}{\text{maximize}} \ \text{var}(\mathbf{Xa}) \ \text{s.t.} \ \|\mathbf{a}\| = 1$$

and is obtained with the ‘rgcca()‘ function as follows:

```
# Design matrix C
# Shrinkage parameters tau = c(tau1, tau2)

pca.with.rgcca = rgcca(A = list(X, X),
                       C = matrix(c(0, 1, 1, 0), 2, 2),
                       tau = c(1, 1))
```

# CCA as RGCCA

Canonical Correlation Analysis is defined as the following optimization problem

$$\underset{\mathbf{a}_1,\mathbf{a}_2}{\text{maximize}} \quad \text{cor}\left(\mathbf{X}_1\mathbf{a}_1, \mathbf{X}_2\mathbf{a}_2\right) \text{ s.t. } \text{var}(\mathbf{X}_1\mathbf{a}_1) = \text{var}(\mathbf{X}_2\mathbf{a}_2) = 1$$

and is obtained with the 'rgcca()' function as follows:

```
# X1 = Block1 and X2 = Block2
# Design matrix C
# Shrinkage parameters tau = c(tau1, tau2)

cca.with.rgcca = rgcca(A= list(X1, X2),
                       C = matrix(c(0, 1, 1, 0), 2, 2),
                       tau = c(0, 0))
```

# PLS as RGCCA

PLS regression is defined as the following optimization problem

$$\underset{\mathbf{a}_1,\mathbf{a}_2}{\text{maximize}} \quad \text{cov}\left(\mathbf{X}_1\mathbf{a}_1, \mathbf{X}_2\mathbf{a}_2\right) \text{ s.t. } \|\mathbf{a}_1\| = \|\mathbf{a}_2\| = 1$$

and is obtained with the ‘rgcca()‘ function as follows:

```
# X1 = Block1 and X2 = Block2
# Design matrix C
# Shrinkage parameters tau = c(tau1, tau2)

pls.with.rgcca = rgcca(A= list(X1, X2),
                       C = matrix(c(0, 1, 1, 0), 2, 2),
                       tau = c(1, 1))
```

# RDA as RGCCA

Redundancy Analysis of $\mathbf{X}_1$ with respect to $\mathbf{X}_2$ is defined as the following optimization problem

$$\underset{\mathbf{a}_1,\mathbf{a}_2}{\text{maximize}} \quad \text{cor}\left(\mathbf{X}_1\mathbf{a}_1, \mathbf{X}_2\mathbf{a}_2\right) \times \text{var}\left(\mathbf{X}_1\mathbf{a}_1\right)^{1/2} \text{ s.t. } \|\mathbf{a}_1\| = \text{var}(\mathbf{X}_2\mathbf{a}_2) = 1$$

and is obtained with the 'rgcca()' function as follows:

```
# X1 = Block1 and X2 = Block2
# Design matrix C
# Shrinkage parameters tau = c(tau1, tau2)

ra.with.rgcca = rgcca(A= list(X1, X2),
                      C = matrix(c(0, 1, 1, 0), 2, 2),
                      tau = c(1, 0))
```

# GCCA as RGCCA

For Generalized Canonical Correlation Analysis (GCCA), a superblock $\mathbf{X}_{J+1} = [\mathbf{X}_1, \ldots, \mathbf{X}_J]$ defined as the concatenation of all the blocks is introduced. GCCA is defined as the following optimization problem

$$\underset{\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_J}{\text{maximize}} \sum_{j=1}^{J} \text{cor}^2(\mathbf{X}_j\mathbf{a}_j, \mathbf{X}_{J+1}\mathbf{a}_{J+1}) \text{ s.t } \text{var}(\mathbf{X}_j\mathbf{a}_j) = 1, j = 1, \ldots, J+1$$

and is obtained with the ‘rgcca()‘ function as follows:

```
# X1 = Block1, ..., XJ = BlockJ, X_{J+1} = [X1, ..., XJ]
# (J+1)*(J+1) Design matrix C
C = matrix(c(0, 0, 0, ..., 0, 1,
             0, 0, 0, ..., 0, 1,
                    ...
             1, 1, 1, ..., 1, 0), J+1, J+1)
# Shrinkage parameters tau = c(tau1, ...,  tauJ, tau_{J+1})
gcca.with.rgcca = rgcca(A= list(X1, ..., XJ, cbind(X1, ..., XJ)),
                        C = C, tau = rep(0, J+1),
                        scheme = "factorial")
```

# MCIA as RGCCA

For Multiple Co-Inertia Analysis (MCIA) a superblock $\mathbf{X}_{J+1} = [\mathbf{X}_1, \ldots, \mathbf{X}_J$ defined as the concatenation of all the blocks is introduced. MCIA is defined as the following optimization problem

$$\underset{\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_J}{\text{maximize}} \sum_{j=1}^{J} \text{cor}^2(\mathbf{X}_j\mathbf{a}_j, \mathbf{X}_{J+1}\mathbf{a}_{J+1}) \times \text{var}(\mathbf{X}_j\mathbf{a}_j), \quad \text{s.t } \|\mathbf{a}_j\| = 1, j = 1, \ldots, J \text{ and} \tag{1}$$

and is obtained with the 'rgcca()' function as follows:

```
# X1 = Block1, ..., XJ = BlockJ, X_{J+1} = [X1, ..., XJ]
# (J+1)*(J+1) Design matrix C
C = matrix(c(0, 0, 0, ..., 0, 1,
             0, 0, 0, ..., 0, 1,
                ...
             1, 1, 1, ..., 1, 0), J+1, J+1)
# Shrinkage parameters tau = c(tau1, ..., tauJ, tau_{J+1})
mcoa.with.rgcca = rgcca(A= list(X1, ..., XJ, cbind(X1, ..., XJ)),
                        C = C, tau = c(rep(1, J), 0),
                        scheme = "factorial")
```

# RGCCA data analysis

Data must be preprocessed to ensure comparability between variables. Standardization is applied (zero mean and unit variance).

```
library(RGCCA)
load("data/breast_TCGA.RData")
X <- t(breast_multi$RNAseq)
Y <- t(breast_multi$miRNA)
Z <- t(breast_multi$RPPA)
A <- list(rnaseq=X, miRNA=Y, RPPA=Z)
A <- lapply(A, scale)
```

NOTE: A possible strategy is to standardize the variables and then to divide each block by the square root of its number of variables. This two-step procedure leads to $\mathrm{tr}(\mathbf{X}_j^t \mathbf{X}_j) = n$ for each block (i.e. the sum of the eigenvalues of the covariance matrix of $\mathbf{X}_j$ is equal to 1 whatever the block). Such a preprocessing is reached by setting the 'scale' argument to 'TRUE' (default value) in the 'rgcca()' and 'sgcca()' functions.

# RGCCA data analysis

Let us assume that both miRNA and mRNa are affecting the protein levels. Therefore the **C** matrix will be

```
C <- matrix(c(1,0,1,0,1,1,0,0,1), nrow=3, byrow = TRUE)
C
```

```
##      [,1] [,2] [,3]
## [1,]    1    0    1
## [2,]    0    1    1
## [3,]    0    0    1
```

# RGCCA data analysis

RGCCA using the defined design matrix **C**, the factorial scheme ($g(x) = x^2$) and mode B for all blocks (full correlation criterion) is obtained by

```
rgcca.factorial <- rgcca(A, C=C, tau = rep(0, 3),
                         scheme ="factorial", ncomp=c(2,2,2),
                         scale = FALSE, verbose = FALSE)
```

The weight vectors, solution of the optimization problemare obtained as:

```
rgcca.factorial$a # weight vectors
```

# RGCCA data analysis

The block-components are also avalaible as output of ‘rgcca‘. The first components of each block are given by:

```
Y.block <- rgcca.factorial$Y
lapply(Y.block, head)

## [[1]]
##                    comp1      comp2
## TCGA-C8-A12V -1.4043762 -0.5590542
## TCGA-A2-A0ST -1.3471553 -2.3964964
## TCGA-E2-A159 -0.5738528  0.8281707
## TCGA-BH-A0BW -0.2061025  0.4681924
## TCGA-A2-A0SX -0.8960506  0.1448149
## TCGA-AR-A1AI -1.2582255  0.4333885
##
## [[2]]
##                    comp1      comp2
## TCGA-C8-A12V -1.4043119 -0.5584358
## TCGA-A2-A0ST -1.3470213 -2.3960350
## TCGA-E2-A159 -0.5738116  0.8287364
## TCGA-BH-A0BW -0.2062718  0.4694125
## TCGA-A2-A0SX -0.8958721  0.1451569
## TCGA-AR-A1AI -1.2582603  0.4338262
##
## [[3]]
##                    comp1      comp2
```

# RGCCA data analysis

Information about Average Variance Explained (AVE)
- The AVE of block $\mathbf{X}_j$, denoted by AVE($\mathbf{X}_j$), is defined as:

$$\text{AVE}(\mathbf{X}_j) = 1/p_j \sum_{h=1}^{p_j} cor^2(\mathbf{x}_{jh}, \mathbf{y}_j)$$

AVE($\mathbf{X}_j$) varies between 0 and 1 and reflects the proportion of variance captured by $\mathbf{y}_j$.
- For all blocks:

$$\text{AVE(outermodel)} = \left(1/\sum_j p_j\right) \sum_j p_j \text{AVE}(\mathbf{X}_j)$$

- For the inner model:

$$\text{AVE(innermodel)} = \left(1/\sum_{j<k} c_{jk}\right) \sum_{j<k} c_{jk} \text{cor}^2(\mathbf{y}_j, \mathbf{y}_k)$$

# RGCCA data analysis

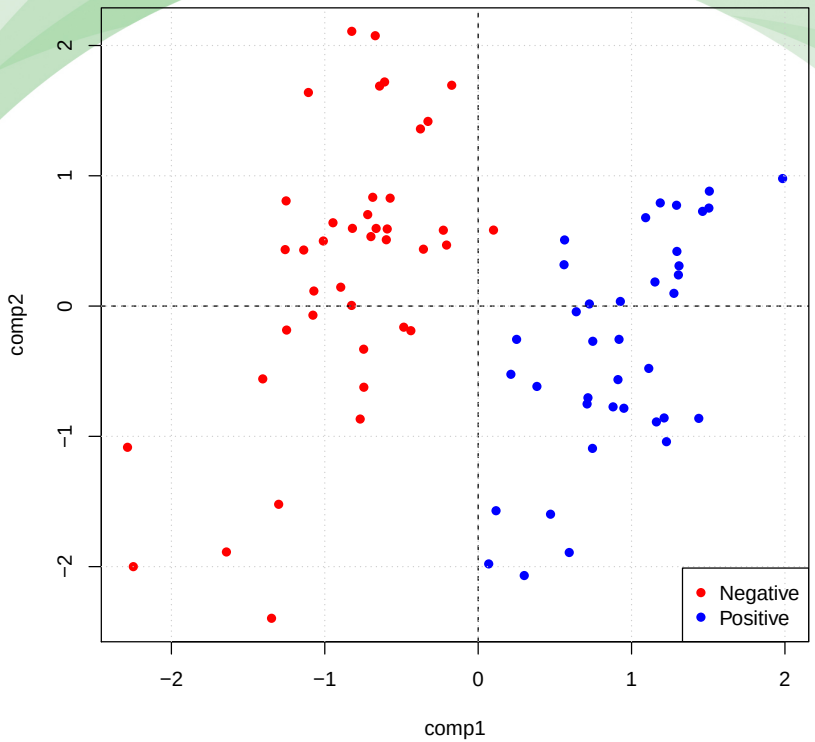
```
rgcca.factorial$AVE

## $AVE_X
## $AVE_X[[1]]
##     comp1     comp2
## 0.1470807 0.0303539
##
## $AVE_X[[2]]
##      comp1      comp2
## 0.06683530 0.02441686
##
## $AVE_X[[3]]
##      comp1      comp2
## 0.17983101 0.07924063
##
##
## $AVE_outer_model
## [1] 0.14088037 0.03060966
##
## $AVE_inner_model
## [1] 1 1
```

# RGCCA data analysis

## Plot individuals

```
source("R/plotInd.R")
plotInd(rgcca.factorial, group)
```

# RGCCA data analysis



# RGCCA data analysis

Features first axis right side (e.g. ER+)

```
source("R/topVars.R")
topVars(rgcca.factorial, axis=1, end="pos", topN=5)

##       table_1   table_2                       table_3
## top_1 "PGLYRP2" "hsa-mir-29b-2.MIMAT0004515" "ER.alpha"
## top_2 "TMEM163" "hsa-mir-29c.MIMAT0004673"   "PR"
## top_3 "CYP4F11" "hsa-mir-190b.MIMAT0004929"  "GATA3"
## top_4 "FDXR"    "hsa-mir-664.MIMAT0005949"   "AR"
## top_5 "WBSCR17" "hsa-mir-342.MIMAT0004694"   "INPP4B"
```

# Outline

1 Introduction
2 Integrating two or more omic datasets
3 RGCCA
4 SGCCA
5 Recommended lectures

# SGCCA data analysis

SGCCA extends RGCCA to address the issue of variable selection. Specifically, RGCCA with all $\tau_j = 1$ equal to 1 is combined with an L1-penalty that gives rise to SGCCA (Tenenhaus et al, Biostatistics, 2014). The SGCCA optimization problem is defined as follows:

$$\underset{\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_J}{\text{maximize}} \sum_{j,k=1}^{J} c_{jk} g(\text{cov}(\mathbf{X}_j\mathbf{a}_j, \mathbf{X}_k\mathbf{a}_k)) \text{ s.t. } \|\mathbf{a}_j\|_2 = 1 \text{ and } \|\mathbf{a}_j\|_1 \leq s_j, j = 1, \ldots,$$

where $s_j$ is a user defined positive constant that determines the amount of sparsity for $\mathbf{a}_j, j = 1, \ldots, J$. The smaller the $s_j$, the larger the degree of sparsity for $\mathbf{a}_j$. The sparsity parameter $s_j$ is usually set based on cross-validation procedures. Alternatively, values of $s_j$ can simply be chosen to result in desired amounts of sparsity.

# SGCCA data analysis

Let us assume we are interested in this analysis (prediction problem)

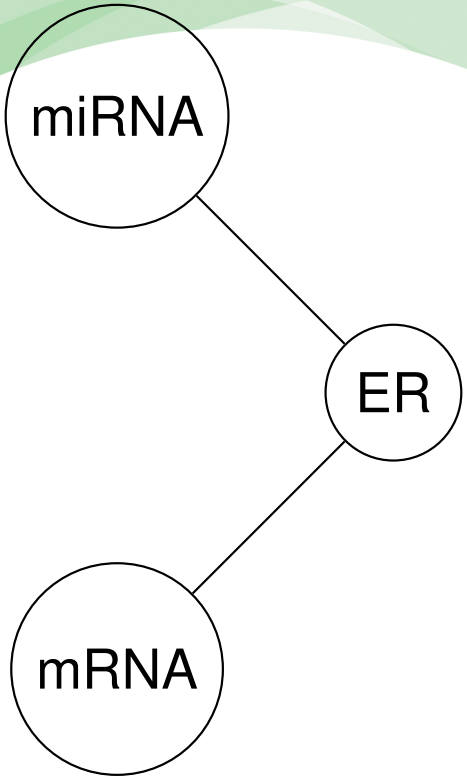



Figure: between-block connection for Breast TCGA data.

# SGCCA data analysis

```
X <- t(breast_multi$RNAseq)
Y <- t(breast_multi$miRNA)
AA <- list(rnaseq=X, miRNA=Y, group=as.numeric(group)-1)
C <-  matrix(c(0, 0, 1, 0, 0, 1, 1, 1, 0), 3, 3)
C
```

```
##      [,1] [,2] [,3]
## [1,]    0    0    1
## [2,]    0    0    1
## [3,]    1    1    0
```

# SGCCA data analysis

Let us estimate the weights for the 1st component

```
sgcca.breast = sgcca(AA, C, c1 = c(.071,.2, 1),
                     ncomp = c(1, 1, 1),
                     scheme = "centroid",
                     scale = TRUE,
                     verbose = FALSE)
```

# SGCCA data analysis

The mRNA associated with ER+ (i.e, positive part of first axis) are (NOTE:results are ordered)

```
source("R/selectVars.R")
ans <- selectVars(sgcca.breast, table=1, axis=1, end="pos")
length(ans)

## [1] 50

ans

##  [1] "CA12"     "FSIP1"    "AGR3"     "ESR1"     "SCUBE2"   "TBC1D9
##  [7] "MLPH"     "FOXA1"    "NAT1"     "C6orf97"  "ABCC8"    "SLC7A8
## [13] "XBP1"     "THSD4"    "INPP4B"   "AGR2"     "C1orf64"  "ABAT"
## [19] "EVL"      "ANXA9"    "IL6ST"    "ERBB4"    "PGR"      "SUSD3"
## [25] "UGCG"     "IGFALS"   "GATA3"    "ACADSB"   "C4orf18"  "NOSTRI
## [31] "SLC39A6"  "TMEM25"   "WFS1"     "STH"      "MAPT"     "KIAA13
## [37] "ENPP1"    "PREX1"    "ABCC11"   "GRPR"     "ANKRD30A" "C14orf
## [43] "JMJD2B"   "C3orf18"  "DNAJC12"  "SIDT1"    "C16orf45" "KRT18"
## [49] "ANKRA2"   "PH-4"
```

# SGCCA data analysis

The miRNA associated with ER+ (e.g. positive part of the first axis) are (NOTE:results are ordered)

```
ans <- selectVars(sgcca.breast, table=2, axis=1, end="pos")
length(ans)

## [1] 5

ans

## [1] "hsa-mir-190b.MIMAT0004929"  "hsa-mir-342.MIMAT0004694"
## [3] "hsa-mir-29b-2.MIMAT0004515" "hsa-mir-29c.MIMAT0004673"
## [5] "hsa-mir-431.MIMAT0004757"
```

# Concluding remarks

Correlation $Y = X$

Multiple correlation $Y = X_1 X_2 X_3 \cdots X_k$

Canonical correlation $Y_1 Y_2 Y_3 \cdots Y_j = X_1 X_2 X_3 \cdots X_k$

- Co-inertia analysis (CIA) is similar to CCA but it optimizes the squared covariance between the eigenvectors while CC optimizes the correlation.
- CIA can be applied to datasets where the number of variables (genes) far exceeds the number of samples (arrays) such is the case in several omic data, while CCA requires a regularized version to be implemented.
- Sparse and reguralized methods requires a tuning parameter. This makes these methods computing demanding.

# Take home messages

- Multivariate methods are purely descriptive methods that do not test a hypothesis to generate a p-value.
- They are not optimized for variable of biomarkers discovery, though the introduction of sparsity in variable loadings may help in the selection of variables forn downstream analyses.
- Number of variables in omic data is a challenge to tradicional visualization tools. New `R` packages including `ggord` are being developed to address this issue.
- Dynamic visualization is possible using `ggvis`, `ploty`, `explor` and other packages.
- Projection in the same space of variable annotation (GO or Reactome) may help to determine gene sets or patwhays associated with our traits.

# Outline

1 Introduction
2 Integrating two or more omic datasets
3 RGCCA
4 SGCCA
5 Recommended lectures

# Recommended lectures


- Millstein et al. (2009). Disentangling molecular relationships with a causal inference test. BMC Bioinf, 10:23.
- Ebrahim and Smith (2008). Mendelian randomization: can genetic epidemiology help redress the failure of observational epidemiology?. Hum Genet, 123:15-33.
- Liu et al. (2013). Epifenome-wide association data implicate DNA methylation as an intermediary of genetic risk in rheumatoid arthritis. Nat Biotech, 31(2):142-148.
- Voight et al. (2012). Plasma HDL cholesterol and risk of myocardial infarction: a mendelian randomisation study. Lancet, 380(9841): 572-580.
- Meng et al. (2014). A multivariate approach to the integration of multi-omics datasets. BMC Bioinf, 15:162.
- Witten et al. (2009). A Penalized Matrix Decomposition, with Applications to Sparse Principal Components and Canonical Correlation Analysis. Biostatistics, 10(3):515-34.
- Meng et al. (2016). Dimension reduction techniques for the integrative analysis of multi-omics data. Brief Bioinform, 17(4): 628-641.
- Tenenhaus A. and Tenenhaus M. (2011). Regularized Generalized Canonical Correlation Analysis. Psychometrika, 76: 257-84.
- Tenenhaus et al. (2014). Variable Selection for Generalized Canonical Correlation Analysis. Biostatistics 15(3):569-83.