THE UNIVERSITY OF BRITISH COLUMBIA
Department of Anesthesiology, Pharmacology & Therapeutics
Faculty of Medicine

Centre for Heart Lung Innovation
UBC and St. Paul's Hospital

CompBio Lab @ HLI
cbl-hli.med.ubc.ca/

Getting started with mixOmics

Amrit Singh, PhD
Assistant Professor

April 13th, 2023 | 11:45-12:35 EST

Comp Bio lab
code

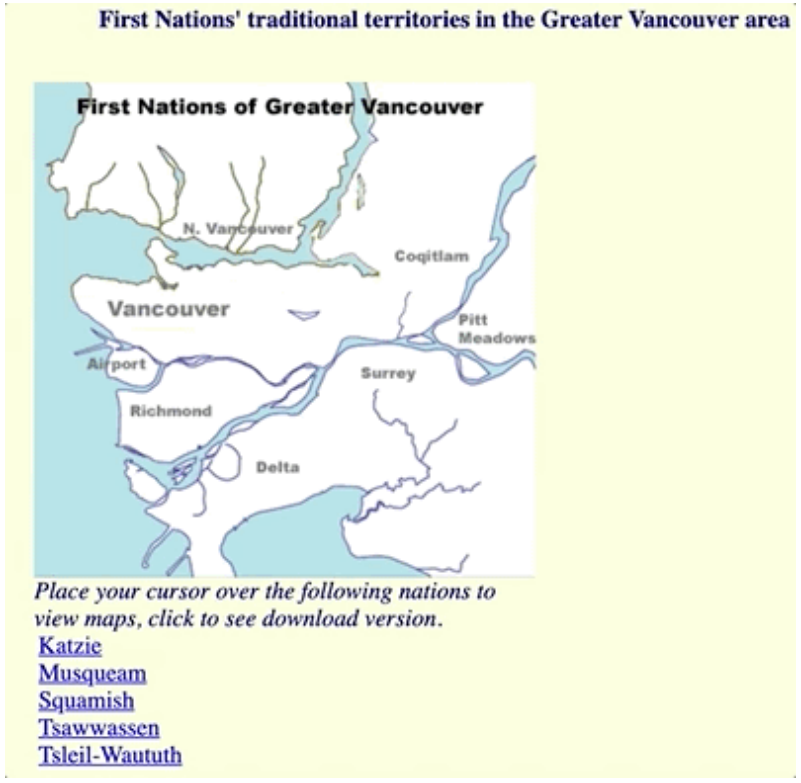
Land acknowledgement

I would like to acknowledge that I work on the traditional, ancestral, and unceded territory of the Coast Salish Peoples, including the territories of the xwməθkwəy̓əm (Musqueam), Sḵwx̱wú7mesh (Squamish), Stó:lō and Səl̓ílwətaʔ/Selilwitulh (Tsleil- Waututh) Nations.

Traditional: Traditionally used and/or occupied by Musqueam people

Ancestral: Recognizes land that is handed down from generation to generation

Unceded: Refers to land that was not turned over to the Crown (government) by a treaty or other agreement



Learning outcomes

1. why mixomics?

2. setting up R, RStudio, and mixOmics

3. Case studies

High-dimensional data

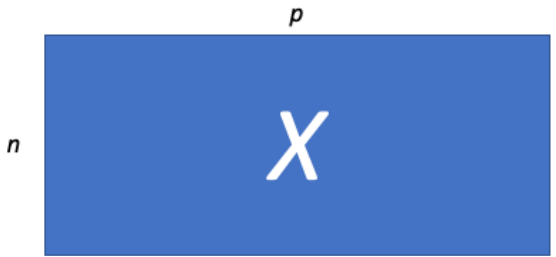
- n <<< p (number of observations is much smaller than the number variables)
- data is highly correlated

univariate

```
##       p_1
## 1   1.030
## 2   1.140
## 3   0.155
## 4  -0.218
## 5  -1.280
## 6   0.248
## 7   1.410
## 8  -0.960
## 9   1.200
## 10  0.025
```

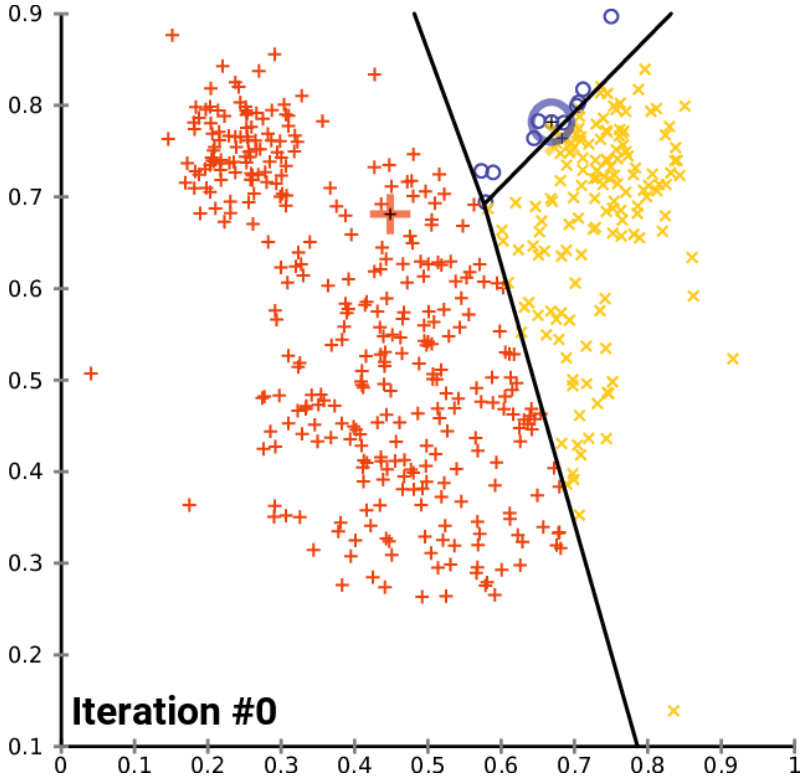
multivariate

```
##       p_1     p_2     p_3    p_4     p_5    p_6     p_7     p_8    p_9   p_10
## 1  -0.953 -0.7780  0.8970  1.080  0.1020  1.400  0.8610 -1.3600 -0.614  0.526
## 2   1.360 -0.7720 -0.3160 -0.150 -0.1740 -0.533 -0.5450 -0.2360  0.495  0.719
## 3   1.400  0.7070  0.0795 -2.090  0.6490  2.210 -1.2100  2.0900  0.226 -0.257
## 4   0.121  2.1500  0.6120  1.150  0.3810 -0.968  0.9220 -0.6310  0.909  0.639
## 5   0.442 -2.2300  0.8610 -0.490  1.5600  0.426  0.0135 -1.2500 -0.649 -0.209
## 6   0.437 -0.1180  0.1390 -1.660 -0.4940 -1.350 -1.7200 -0.2710 -0.546 -1.020
## 7  -1.700 -0.8210  1.4400 -0.456 -0.9030  0.767  0.8400 -0.0561  0.604  1.440
## 8   0.218 -0.0813 -0.3650 -0.495 -0.7760  0.152  1.4200 -0.5030 -1.060 -0.578
## 9  -1.630 -0.7870 -1.4900 -0.407  0.0516 -1.230  0.2610  0.2690 -1.340 -0.514
## 10  0.508 -0.3870 -0.9030  0.134  0.6480 -0.832 -1.7500 -0.8150  0.861 -0.617
```



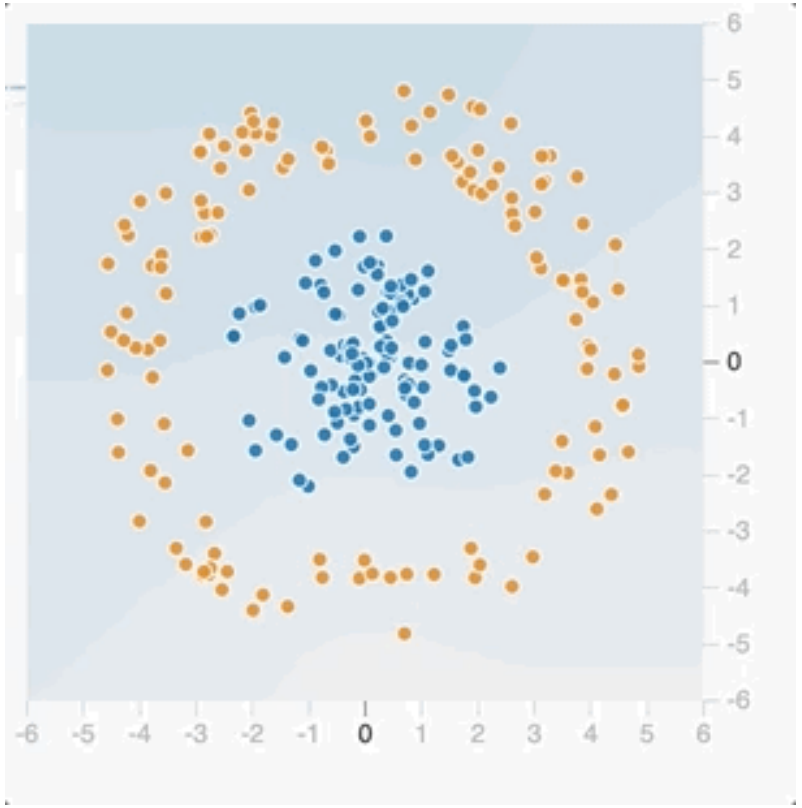
What can you do with high-dimensional data?

Unsupervised (clustering)



Chire 2017

Supervised (regression/classification)



Tensorflow playground

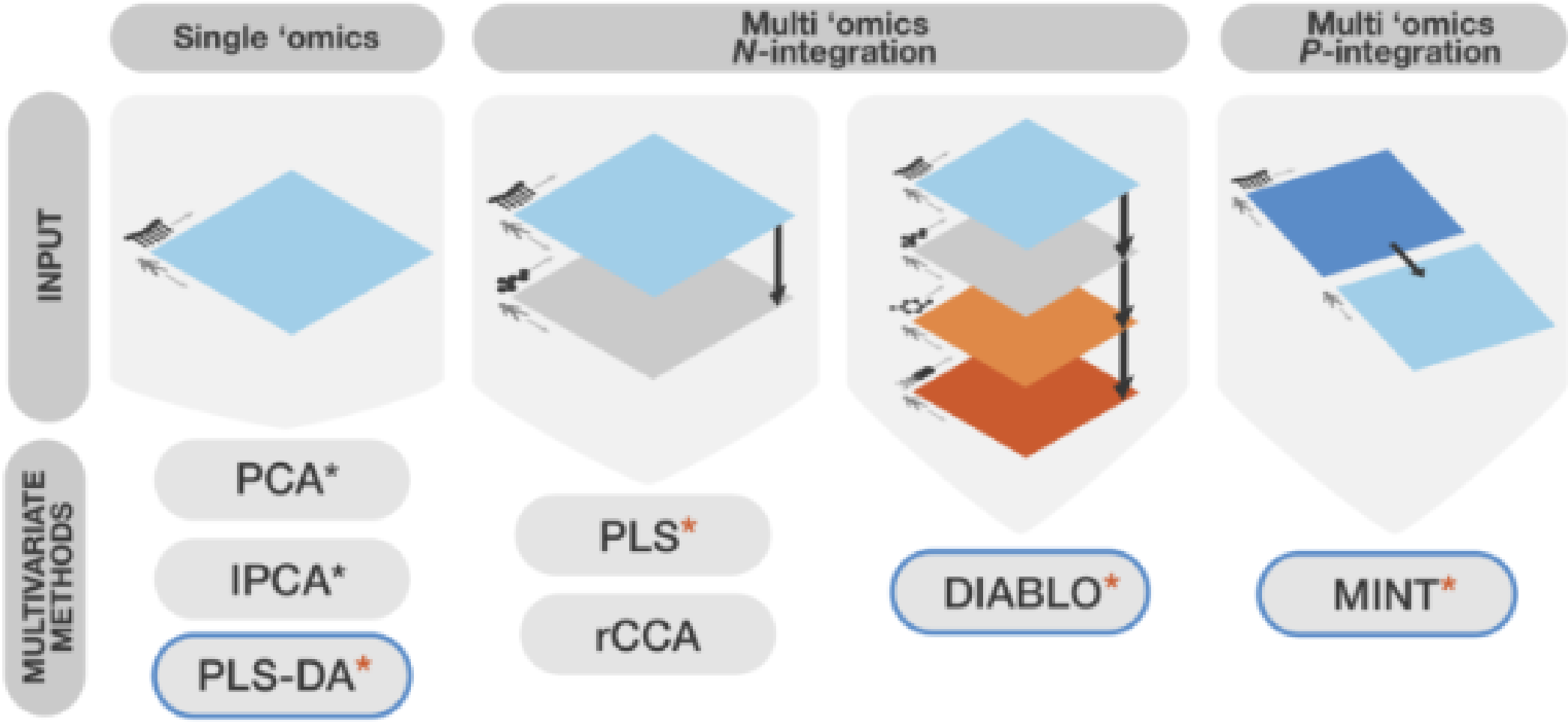
mixOmics

- initiative started and maintained by Prof Kim-Anh Lê Cao
- R-library with 19 methods for high-dimensional data (exploratory analyses, classification, regression, data integration, meta-analysis)

Lab head: A/Prof Kim-Anh Lê Cao
NHMRC Career Development Fellow
Melbourne Integrative Genomics (MIG) & School of Mathematics and Statistics
Building 184 ground floor | University of Melbourne | Parkville VIC 3010

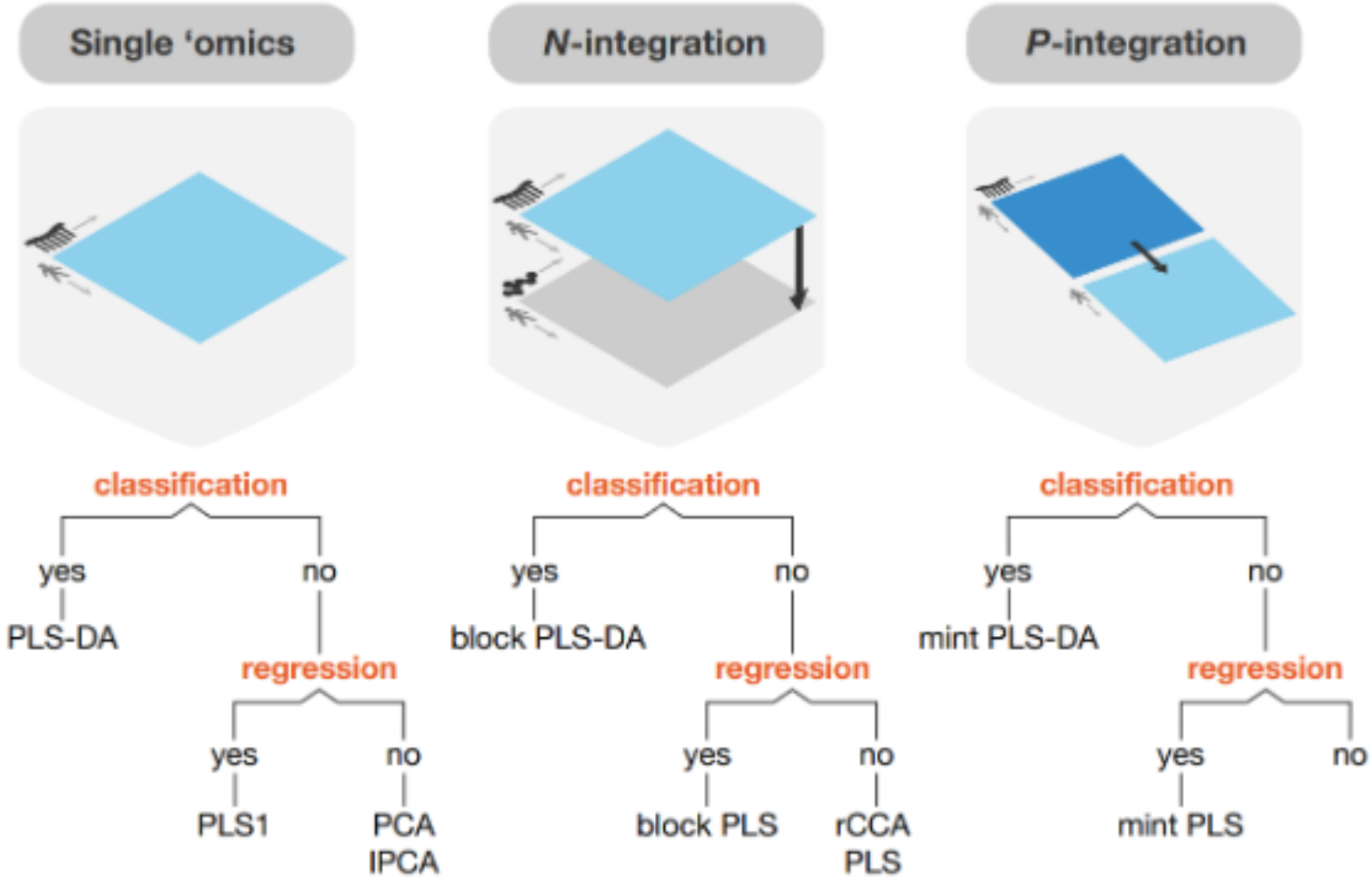
@: kimanh.lecao[at]unimelb.edu.au | twitter: @mixOmics_team | Ph: +61 3 8344 3971

What does mixOmics offer? methods...

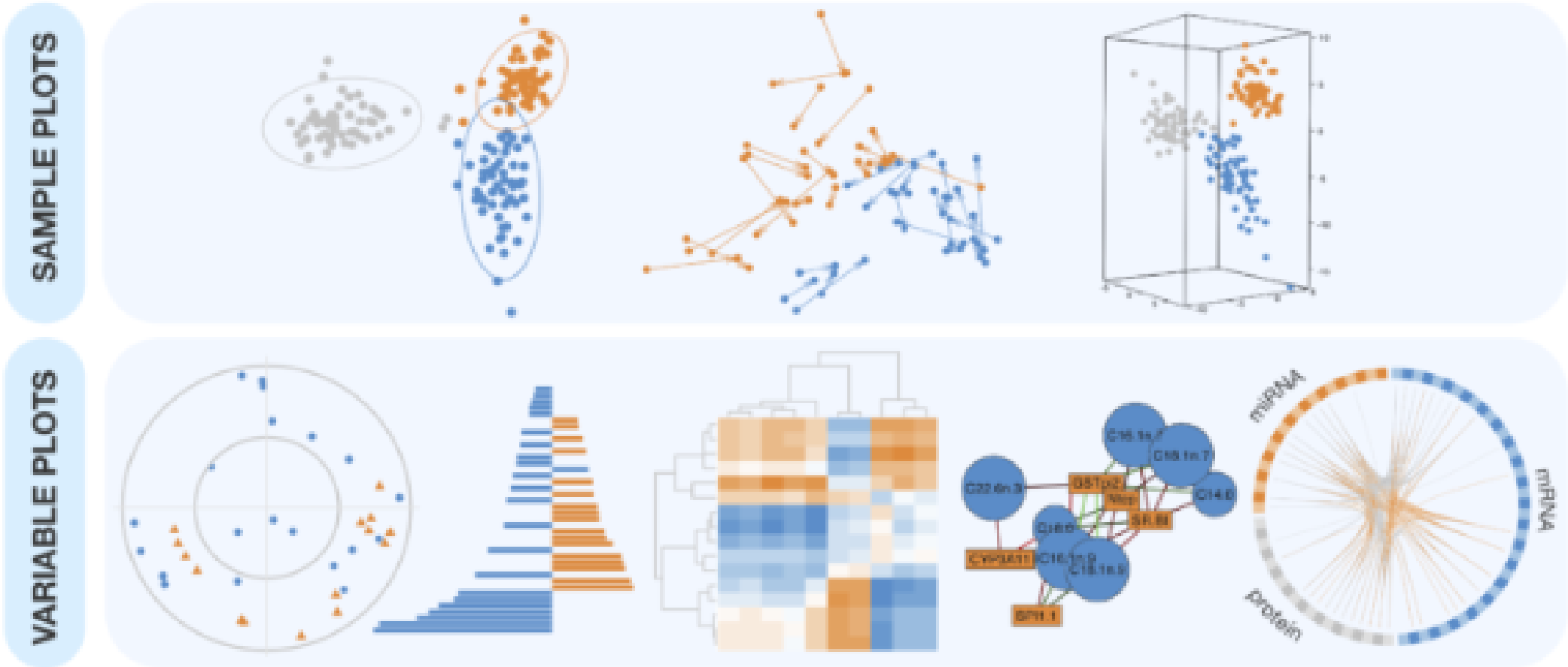


mixOmics.org | *variable selection

What does mixOmics offer? when to use these methods...



What does mixOmics offer? graphics...



mixOmics.org

Getting started with mixOmics

1. Download R
2. Download RStudio
3. install mixOmics

install mixOmics

```
if (!require("BiocManager", quietly = TRUE))
    install.packages("BiocManager")

BiocManager::install("mixOmics")
```

load vignette

```
openVignette("mixOmics")
```

Dataset used in this talk

Breast Cancer multi omics data from TCGA

This data set is a small subset of the full data set from The Cancer Genome Atlas that can be analysed with the DIABLO framework. It contains the expression or abundance of three matching omics data sets: mRNA, miRNA and proteomics for 150 breast cancer samples (Basal, Her2, Luminal A) in the training set, and 70 samples in the test set. The test set is missing the proteomics data set.

```
library(mixOmics)
data(breast.TCGA)
lapply(breast.TCGA$data.train, dim)
```

```
## $mirna
## [1] 150 184
##
## $mrna
## [1] 150 200
##
## $protein
## [1] 150 142
##
## $subtype
## NULL
```

breast cancer subtypes

```
addmargins(table(breast.TCGA$data.train$subtype))
```

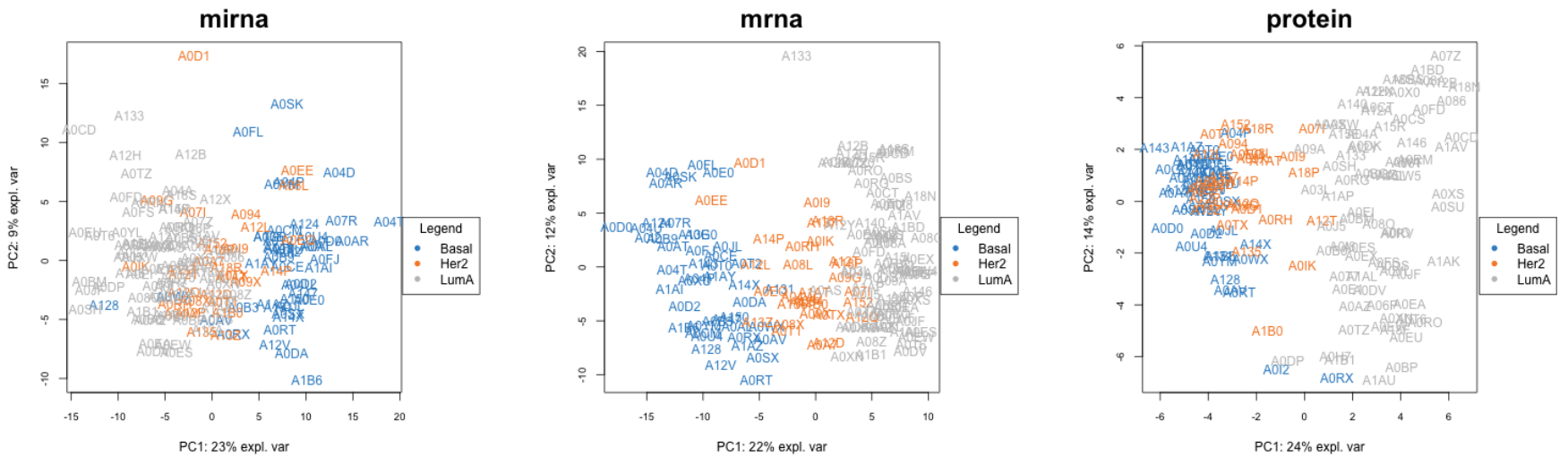
```
##
## Basal  Her2  LumA   Sum
##    45    30    75   150
```

Analysis plan

Analysis	Methods	Functions
Exploratory data analysis	PCA	pca() plotIndiv()
Discriminant analysis	sPLSDA	splsda() tune(), perf() plotIndiv(), plotVar()
Data integration analysis	DIABLO	block.splsda() tune(), perf() plotDiablo(), circosPlot()

Exploratory data analysis using PCA

```
J <- length(breast.TCGA$data.train)-1
pcs <- lapply(breast.TCGA$data.train[1:J], pca)

mapply(function(pca, dataset){
  plotIndiv(pca,
            title=dataset,
            group=breast.TCGA$data.train$subtype,
            style="graphics",
            legend=TRUE)
}, pca=pcs, dataset=names(breast.TCGA$data.train)[1:J])
```



Discriminant analysis using sPLSDA

- based on the eda it seems **mrna** is better at separating classes than **mirna**, lets test this.
- this may or may not be true since we peeked at the data (need to test model with another dataset)

- mrna

```
mrna_model <- splsda(X = breast.TCGA$data.train$mrna,
                     Y = breast.TCGA$data.train$subty
                     keepX = c(5, 5),
                     ncomp = 2)
mrna_perf <- perf(mrna_model, validation = "Mfold", fo
mrna_perf$error.rate
```

```
## $overall
##        max.dist centroids.dist mahalanobis.dist
## comp1 0.2186667      0.1706667        0.1706667
## comp2 0.1000000      0.0960000        0.1213333
##
## $BER
##        max.dist centroids.dist mahalanobis.dist
## comp1 0.3487407      0.1866667        0.1866667
## comp2 0.1281481      0.1085926        0.1376296
```

- mirna

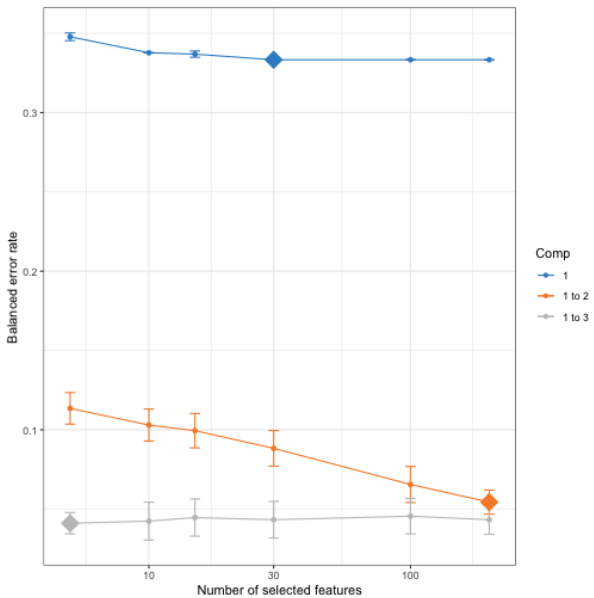
```
mirna_model <- splsda(X = breast.TCGA$data.train$mirna
                      Y = breast.TCGA$data.train$subty
                      keepX = c(5, 5),
                      ncomp = 2)
mirna_perf <- perf(mirna_model, validation = "Mfold",
mirna_perf$error.rate
```

```
## $overall
##        max.dist centroids.dist mahalanobis.dist
## comp1 0.2506667      0.2920000            0.292
## comp2 0.2173333      0.2306667            0.204
##
## $BER
##        max.dist centroids.dist mahalanobis.dist
## comp1 0.3777778      0.3226667        0.3226667
## comp2 0.2979259      0.2549630        0.2392593
```

Find optimal models

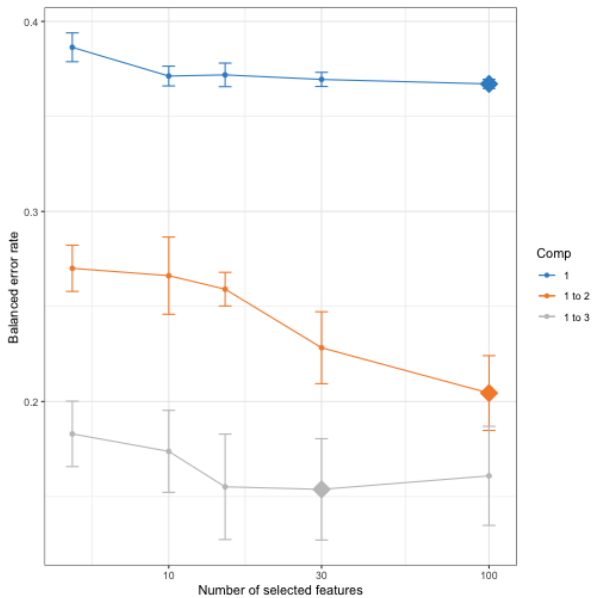
- mrna

```
tune_mrna = tune(method = "splsda", X = breast.TCGA$da
                 Y=breast.TCGA$data.train$subtype, nco
                 test.keepX = c(5, 10, 15, 30, 100, 20
                 progressBar = FALSE)

plot(tune_mrna)
```



- mirna

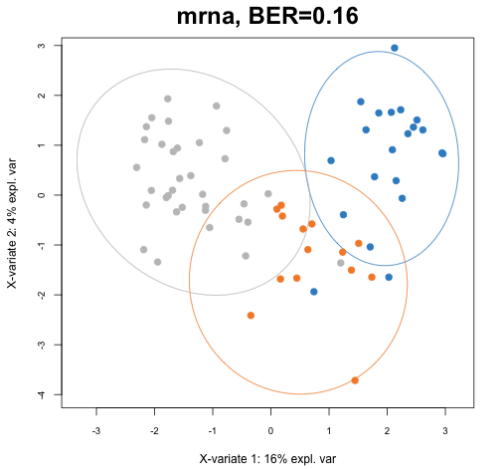
```
tune_mirna = tune(method = "splsda", X = breast.TCGA$d
                  Y=breast.TCGA$data.train$subtype, nco
                  test.keepX = c(5, 10, 15, 30, 100, 20
                  progressBar = FALSE)

plot(tune_mirna)
```



Test sPLSDA models using data from other observations (patients)

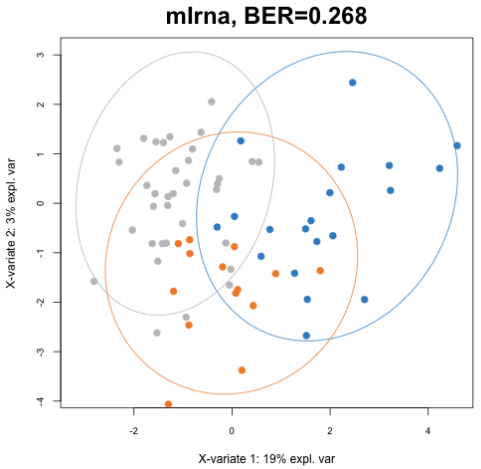
- mrna

```
mrna_model <- splsda(X = breast.TCGA$data.train$mrna,
                     Y = breast.TCGA$data.train$subty
                     keepX = rep(5, 3),
                     ncomp = 3)
mrna_pred <- predict(mrna_model, newdata = breast.TCGA

plotIndiv(mrna_model, comp = 1:2, rep.space = "X-varia
points(mrna_pred$variates[, 1], mrna_pred$variates[, 2
```



- mirna

```
mirna_model <- splsda(X = breast.TCGA$data.train$mirna
                      Y = breast.TCGA$data.train$subty
                      keepX = rep(5, 3),
                      ncomp = 3)
mirna_pred <- predict(mirna_model, newdata = breast.TC

plotIndiv(mirna_model, comp = 1:2, rep.space = "X-vari
points(mirna_pred$variates[, 1], mirna_pred$variates[,
```

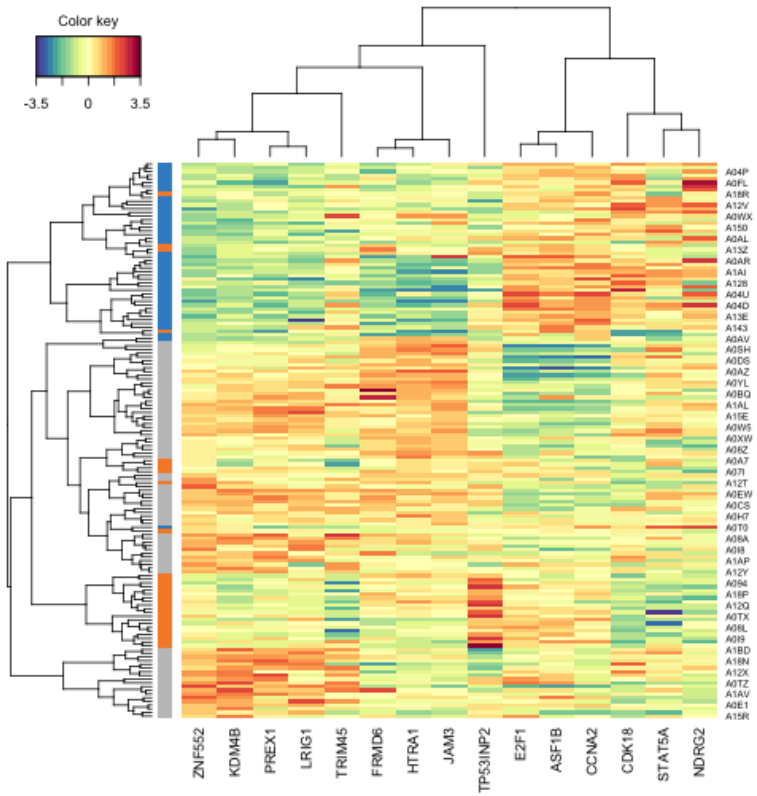


Variables in mrna model

```
rbind(selectVar(mrna_model, comp=1)$value,
      selectVar(mrna_model, comp=2)$value,
      selectVar(mrna_model, comp=3)$value)
```

```
##             value.var
## ZNF552    -0.75801237
## KDM4B     -0.58296361
## PREX1     -0.20979766
## LRIG1     -0.17185790
## CCNA2      0.10963799
## CDK18      0.69045321
## TP53INP2  -0.68042479
## NDRG2      0.22678162
## STAT5A     0.07254351
## TRIM45     0.06003335
## JAM3       0.72727214
## E2F1      -0.53439117
## FRMD6      0.33920140
## ASF1B     -0.23142418
## HTRA1      0.12994836
```

```
cim(mrna_model,
    row.sideColors = mixOmics::color.mixo(as.numeric(b
```



Data integration using DIABLO

Design matters!

```
data = list(mrna = breast.TCGA$data.train$mrna, mirna
            protein = breast.TCGA$data.train$protein)
# set up a full design where every block is connected
# could also consider other weights, see our mixOmics
design = matrix(1, ncol = length(data), nrow = length(
                dimnames = list(names(data), names(dat
diag(design) =  0
design
```

```
##         mrna mirna protein
## mrna       0     1       1
## mirna      1     0       1
## protein    1     1       0
```

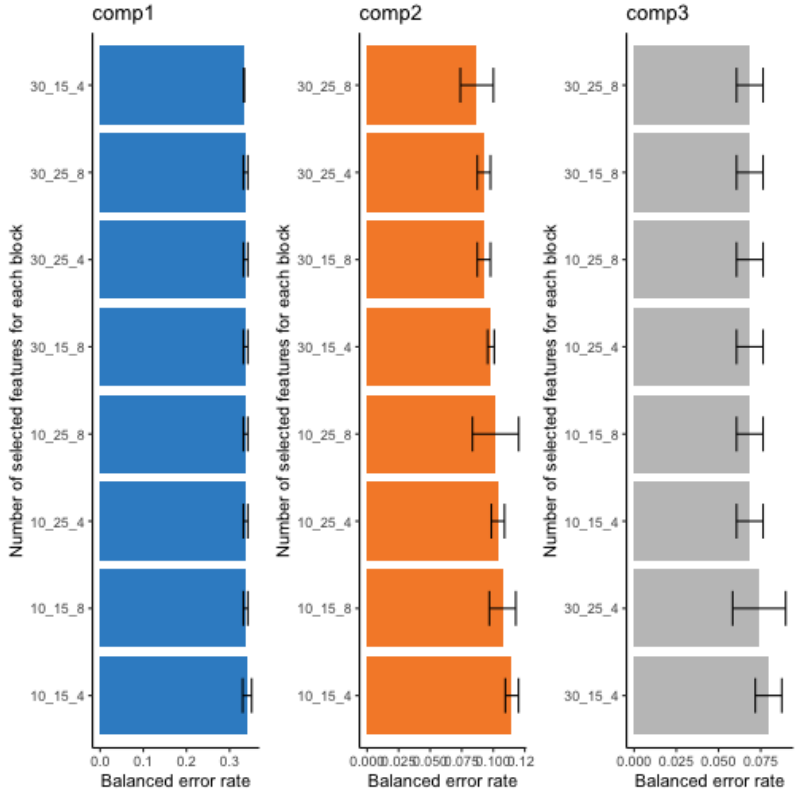
```
# set number of component per data set
ncomp = 3
test.keepX = list(mrna = c(10, 30), mirna = c(15, 25),

## setup cluster - use SnowParam() on Widnows
BPPARAM <- BiocParallel::MulticoreParam(workers = para
tune <- tune.block.splsda(
    X = data,
    Y = breast.TCGA$data.train$subtype,
    ncomp = ncomp,
    test.keepX = test.keepX,
    design = design,
    nrepeat = 2,
    BPPARAM = BPPARAM
)
```

```
## Design matrix has changed to include Y; each block will be
##              linked to Y.

##
## You have provided a sequence of keepX of length: 2 for block
## This results in 8 models being fitted for each component and
```

Bioinformatics. 2019 Sep 1;35(17):3055-3062.

Finding the optimal DIABLO model

```
plot(tune)
```

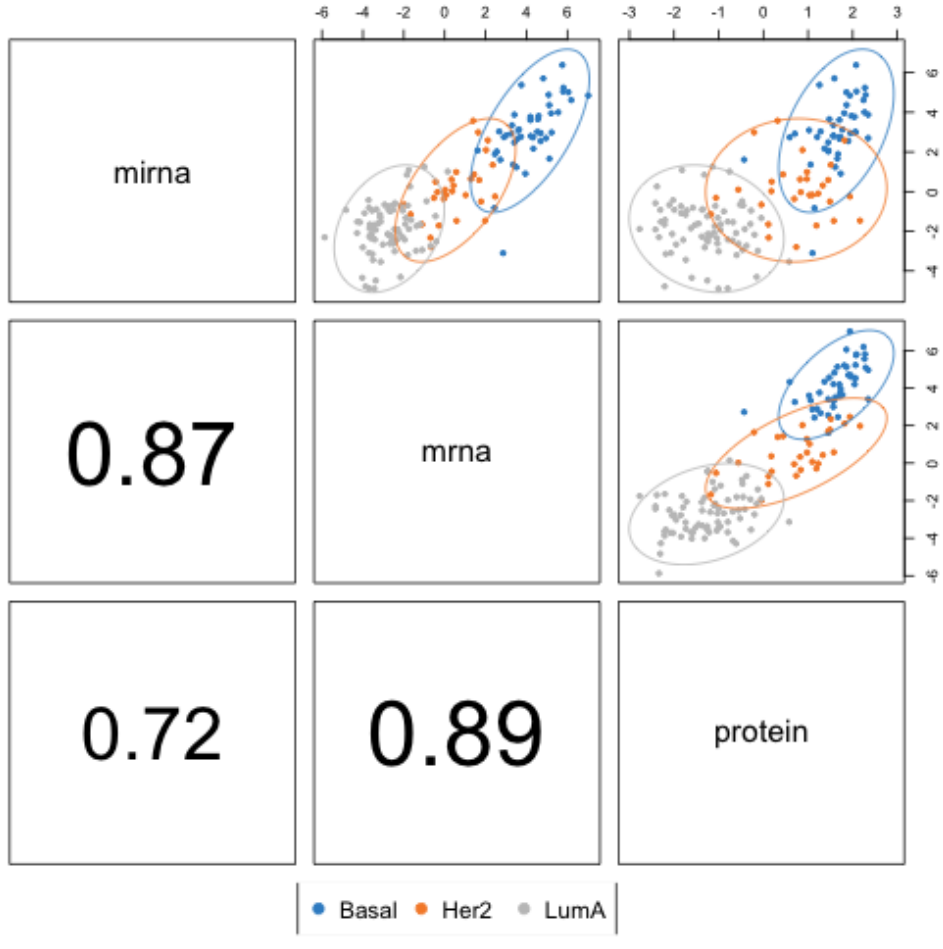



```
tune$choice.keepX
```

```
## $mrna
## [1] 30 30 10
##
## $mirna
## [1] 15 25 15
##
## $protein
## [1] 4 8 4
```

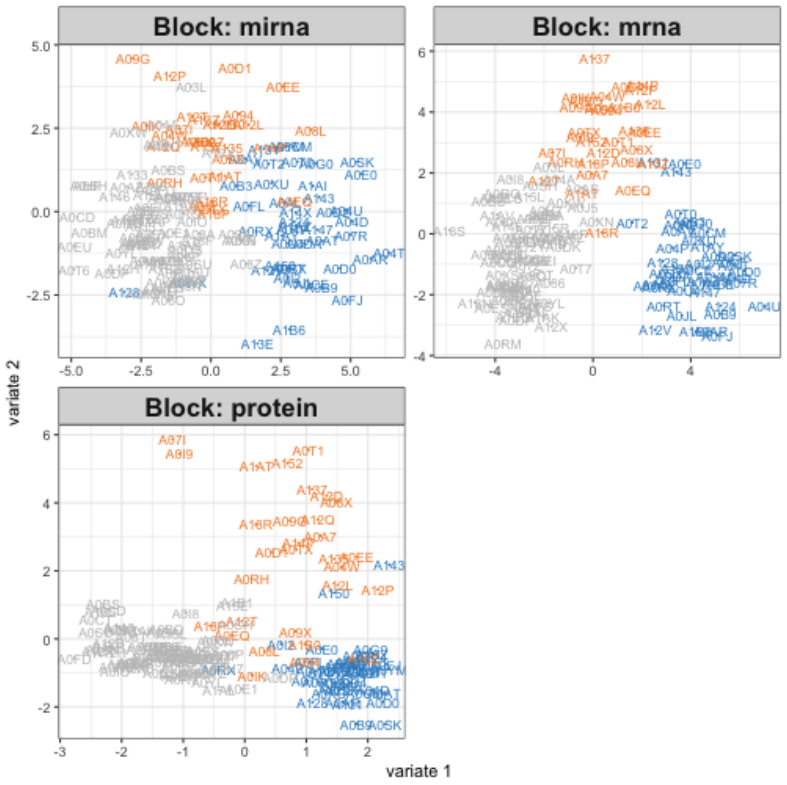
DIABLO model

```
ncomp = length(tune$choice.keepX$mrna)

diablo <- block.splsda(X = breast.TCGA$data.train[1:3]
                       Y = breast.TCGA$data.train$subt
                       keepX=tune$choice.keepX,
                       ncomp = ncomp)
```

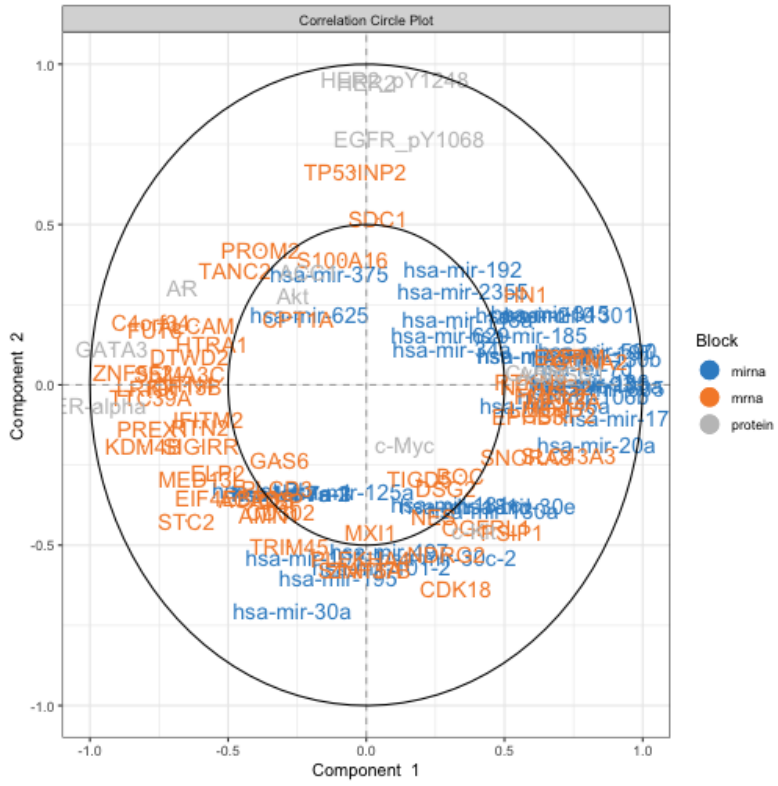


DIABLO: Sample and variable plots

```
plotIndiv(diablo)
```




```
plotVar(diablo, var.names = c(TRUE, TRUE, TRUE),
        legend=TRUE, pch=c(16,16,1))
```

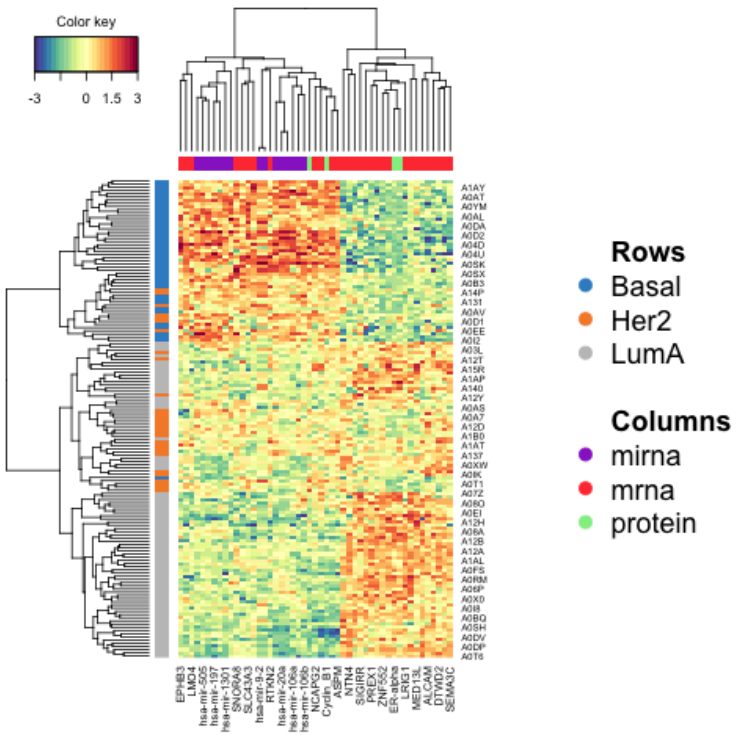


DIABLO

```
cimDiablo(diablo, color.blocks = c('darkorchid', 'brow
          comp = 1, margin=c(8,20), legend.position =
```

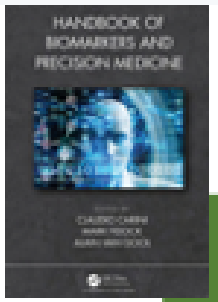
```
network(diablo, blocks = c(1,2,3),
        color.node = c('darkorchid', 'brown1', 'lightg
        cutoff = 0.75)
```

```
##
## trimming values to [-3, 3] range for cim visualisation. See
```



mikelove / awesome-multi-omics Public Watch 45 Fork 123 Star 502

seandavi / awesome-single-cell Public Watch 238 Fork 773 Star 2.2k



Unsupervised

Supervised

Data-driven

Knowledge-based

MCIA: omicade4
sMB-PLS[a]
SGCCA: RGCCA/mixOmics

Multiblock data analyses
Extensions of generalized CCA

SMSMA: msma
SGCCA+DA: mixOmics

SNF:
SNFtool

Message-passing algorithms

PANDA:
pandaR

Joint NMF[a]

Factorization methods

BCC:
bayesCC

iBAG[b]

Bayesian Methods

Bayesian Networks[c]

Concatenation: caret
Ensemble: caretEnsemble

Classification and regression algorithms

SNMNMF[a]

Network-constrained

stSVM: netclass
GELnet: gelnet

Singh A et al., Handbook of Biomarkers and Precision Medicine CRC Press 2019:596

THE UNIVERSITY OF BRITISH COLUMBIA
Department of Anesthesiology, Pharmacology & Therapeutics
Faculty of Medicine

Centre for Heart Lung Innovation
UBC and St. Paul's Hospital

CompBio Lab @ HLI
cbl-hli.med.ubc.ca/

THANK YOU!

April 13th, 2023 | 11:45-12:35 EST

Comp Bio lab
code