Feature Selection and Supervised Omics Integration

Omics Integration and Systems Biology course

Nikolay Oskolkov, Lund University, NBIS SciLifeLab, Sweden

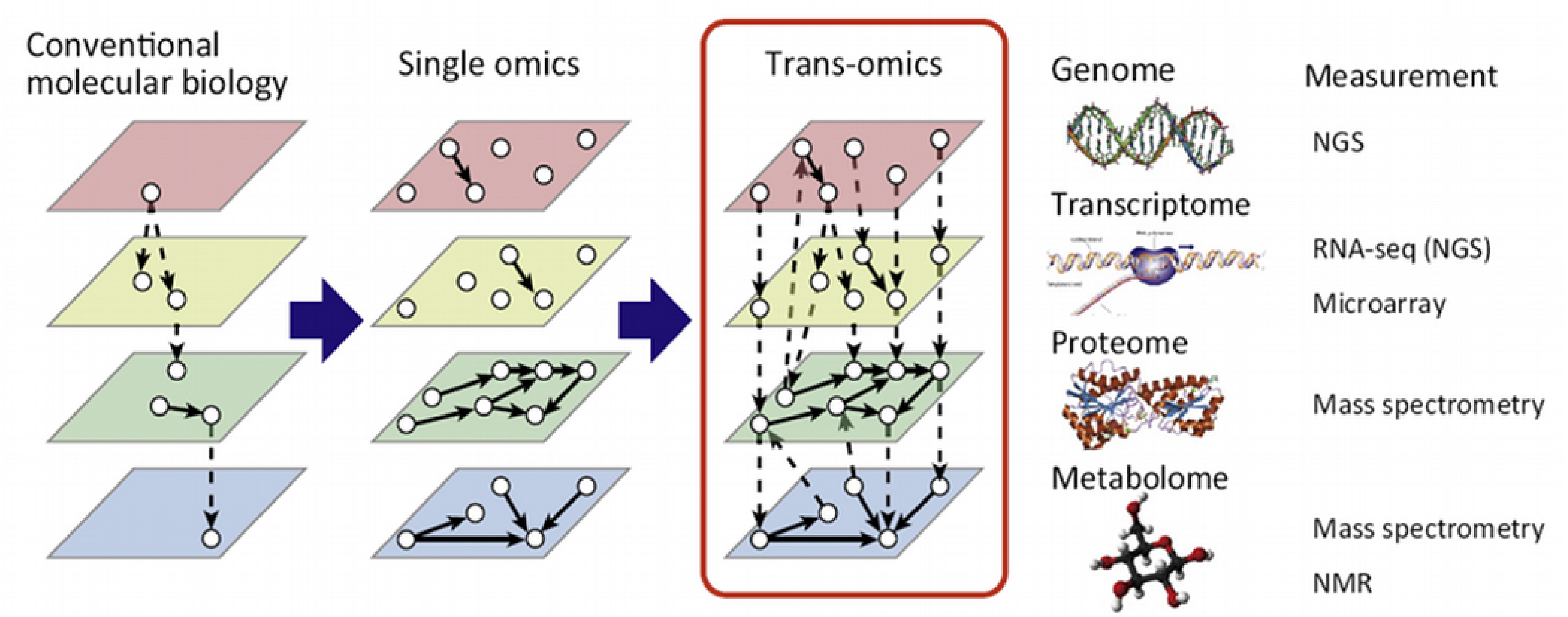


Image adapted from Yugi et al., Trends Biotechnol. 2016

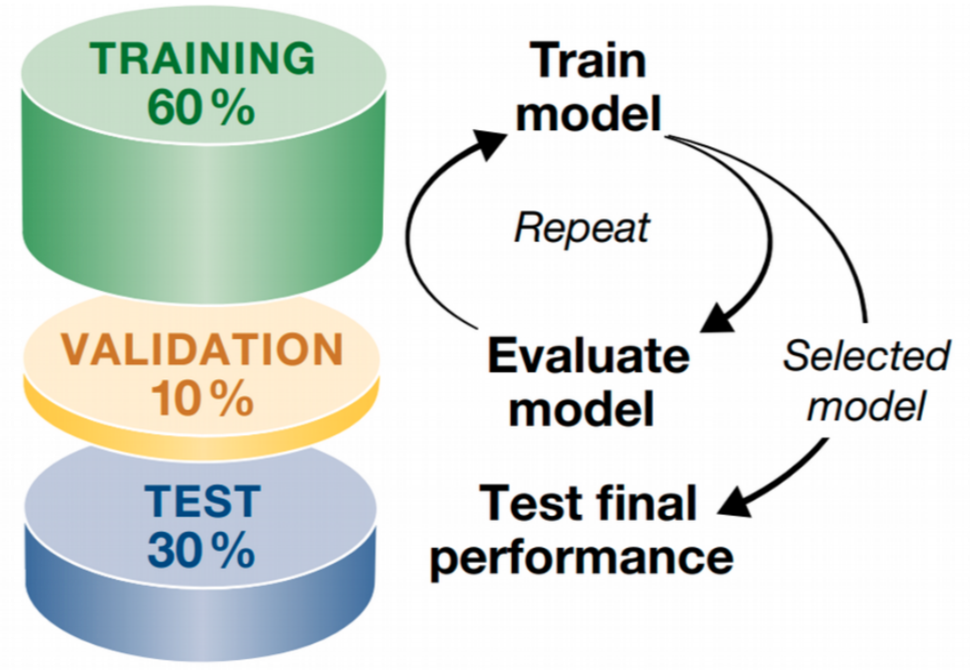
Brief Introduction to Supervised Machine Learning

How does Machine Learning work?

Y = f (X), where X is input (data) and Y is output (response)

Y is present – supervised machine learning

Y is absent – unsupervised machine learning



Machine Learning typically involves five basic steps:

1. Split data set into **train**, **validation** and **test** subsets

2. Fit the model on the train subset

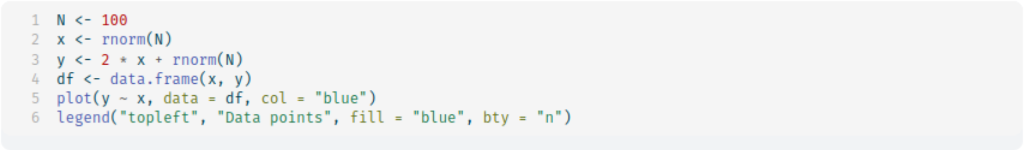
3. Validate your model on the validation subset

4. Repeat train - validation split many times and tune **hyperparameters**

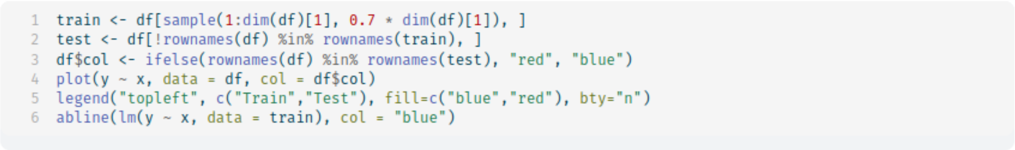
5. Test the accuracy of the optimized model on the test subset.

Image adapted from Angenmueller et al., Mol. Syst. Biol. 2016

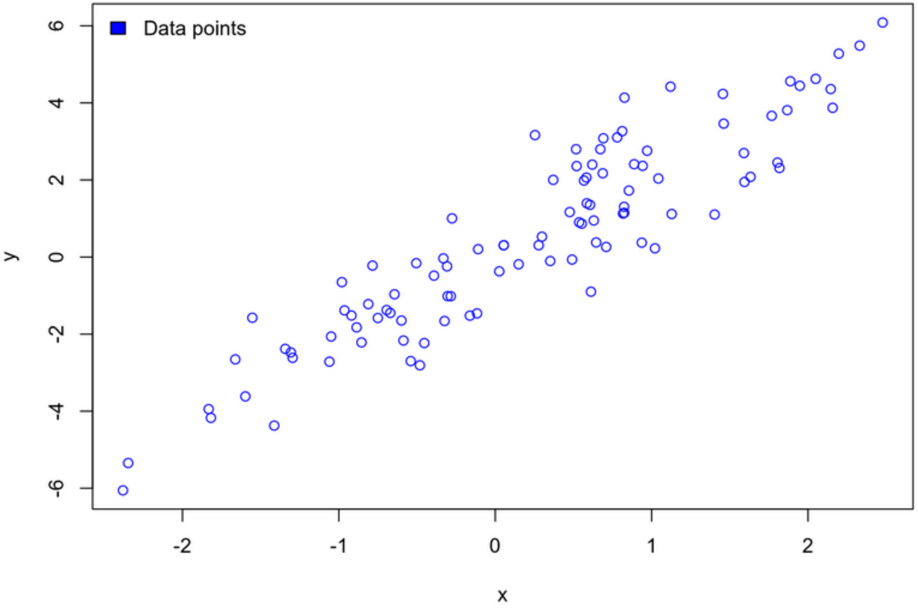
Toy example of supervised machine learning

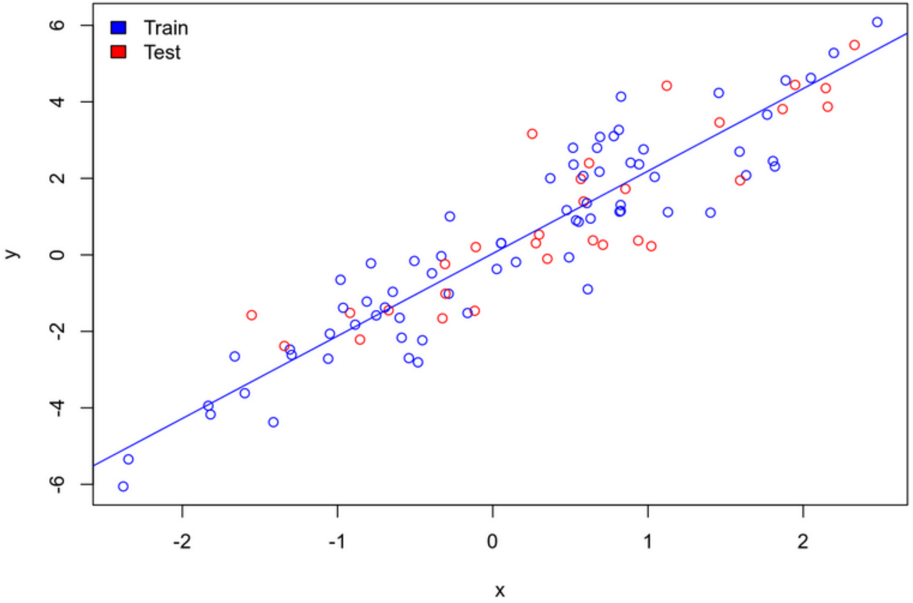


```r
N <- 100
x <- rnorm(N)
y <- 2 * x + rnorm(N)
df <- data.frame(x, y)
plot(y ~ x, data = df, col = "blue")
legend("topleft", "Data points", fill = "blue", bty = "n")
```

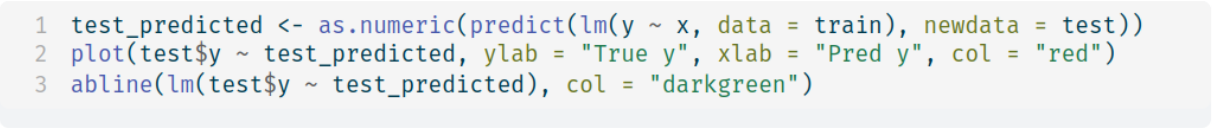


```r
train <- df[sample(1:dim(df)[1], 0.7 * dim(df)[1]), ]
test <- df[!rownames(df) %in% rownames(train), ]
df$col <- ifelse(rownames(df) %in% rownames(test), "red", "blue")
plot(y ~ x, data = df, col = df$col)
legend("topleft", c("Train","Test"), fill=c("blue","red"), bty="n")
abline(lm(y ~ x, data = train), col = "blue")
```

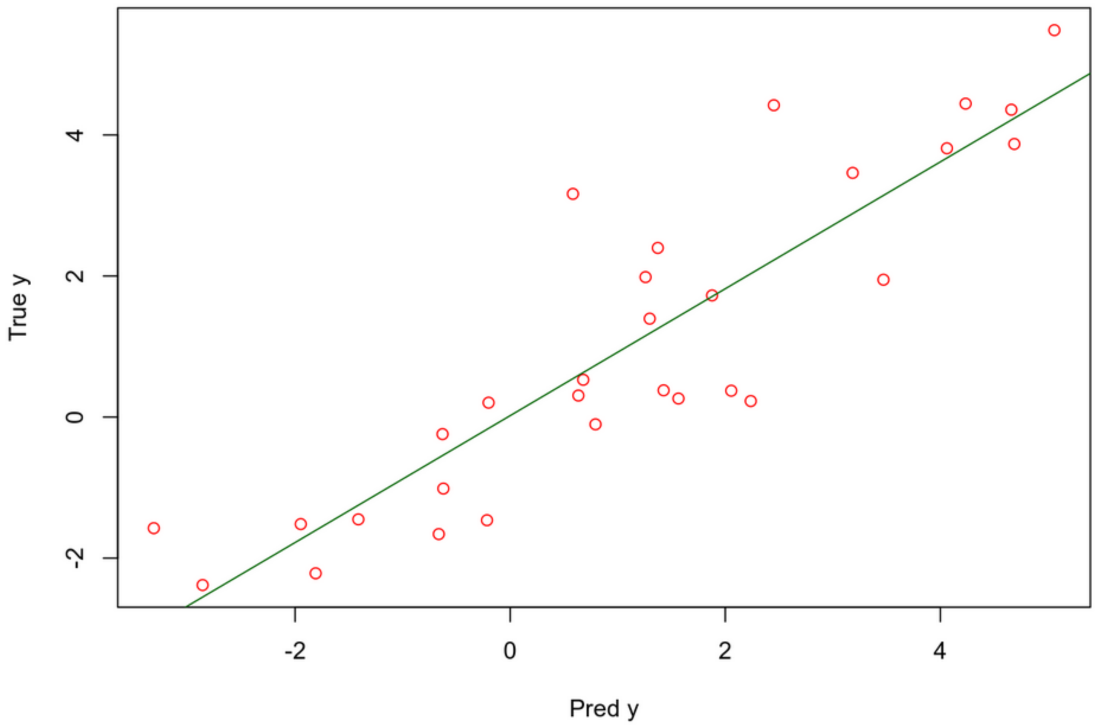




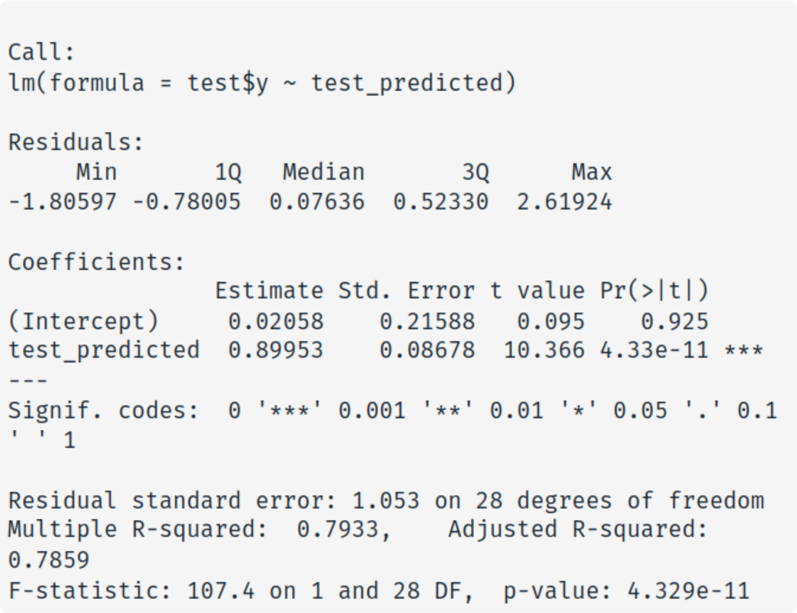
Machine learning model validation



```r
test_predicted <- as.numeric(predict(lm(y ~ x, data = train), newdata = test))
plot(test$y ~ test_predicted, ylab = "True y", xlab = "Pred y", col = "red")
abline(lm(test$y ~ test_predicted), col = "darkgreen")
```



```r
summary(lm(test$y ~ test_predicted))
```



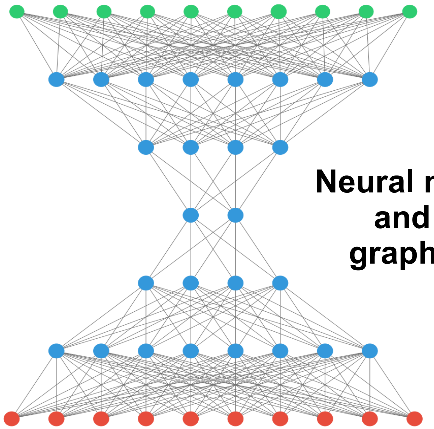
```

Call:
lm(formula = test$y ~ test_predicted)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.80597 -0.78005  0.07636  0.52330  2.61924 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)     0.02058    0.21588   0.095    0.925    
test_predicted  0.89953    0.08678  10.366 4.33e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1

Residual standard error: 1.053 on 28 degrees of freedom
Multiple R-squared:  0.7933,    Adjusted R-squared:
0.7859 
F-statistic: 107.4 on 1 and 28 DF,  p-value: 4.329e-11
```

Thus the model explains 79% of variation on the test subset.

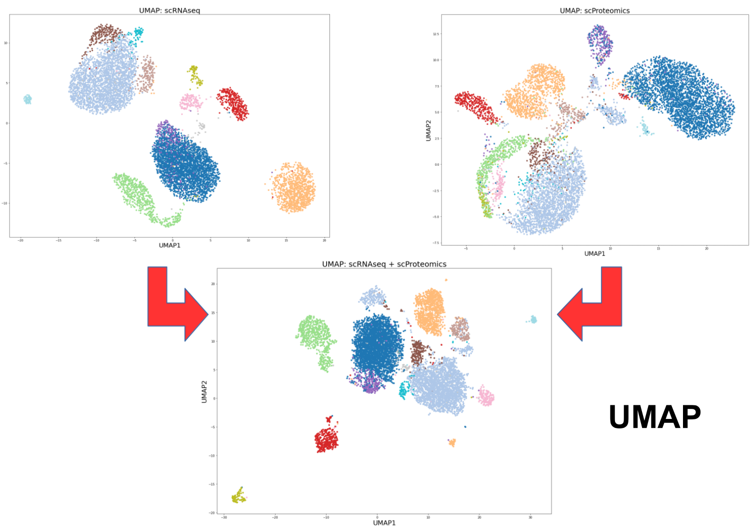
Supervised Machine Learning applied to Omics Integration

Overview of Integrative Methods

Convert to common space

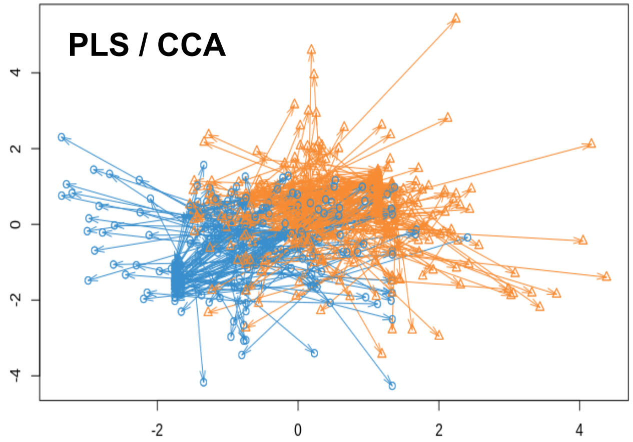


Neural nets and graphs

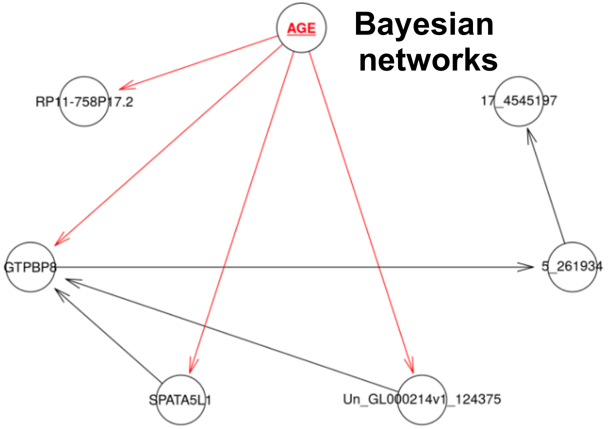


UMAP

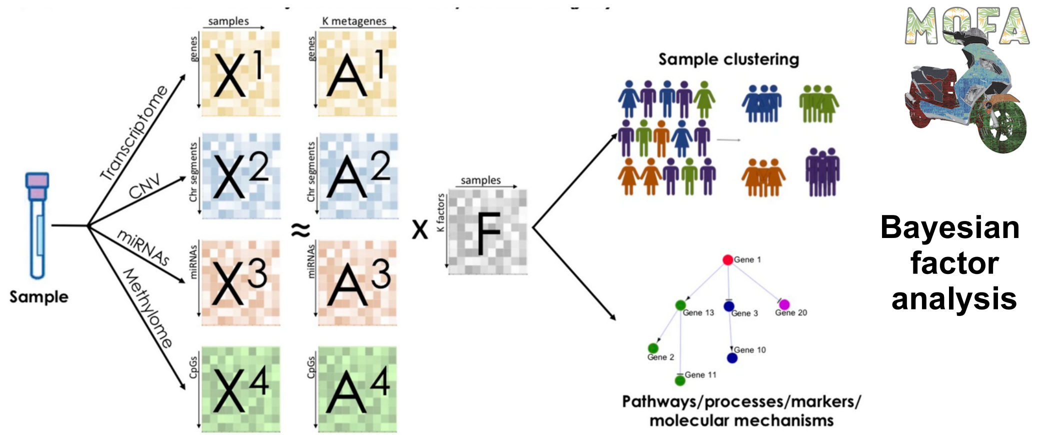
Extract common variation



Combine via Bayes rule



Bayesian networks



Bayesian factor analysis

What Method of Integration to Select?

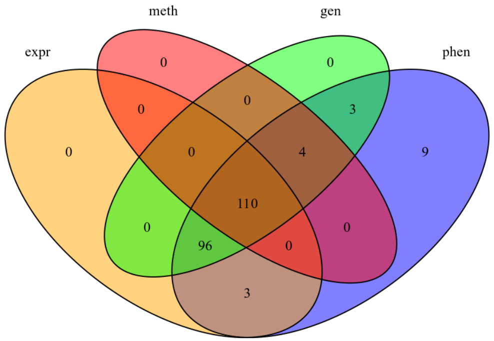
	Linear	Non-Linear
Supervised	PLS / OPLS / mixOmics, LASSO / Ridge / Elastic Net	Neural Networks, Random Forest, Bayesian Networks
Unsupervised	Factor Analysis / MOFA	Autoencoder, SNF, UMAP, Clustering of Clusters

For Example:

1) With ~100 samples it is a good idea to do **linear** Omics integration

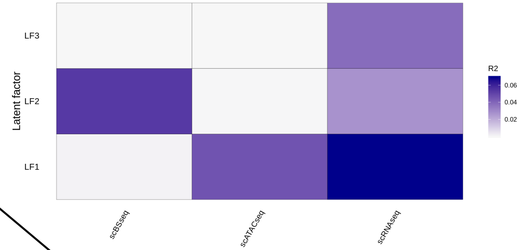
2) T2D is a phenotype of interest, therefore **supervised** integration

Typical Workflow for Omics Integration



Data Set (4 Omics)
110 overlapping individuals

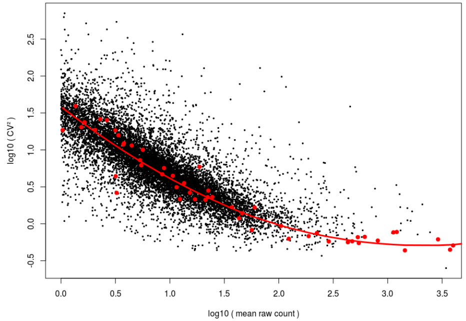
Check covariance



Train Set (n = 80)

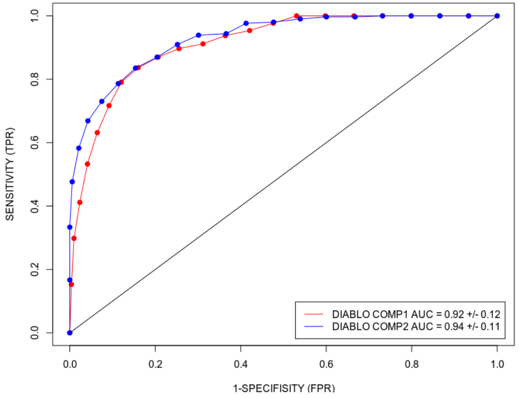
Test Set (n = 30)

Unsupervised:
remove low-variance



Supervised:
LASSO

Evaluation

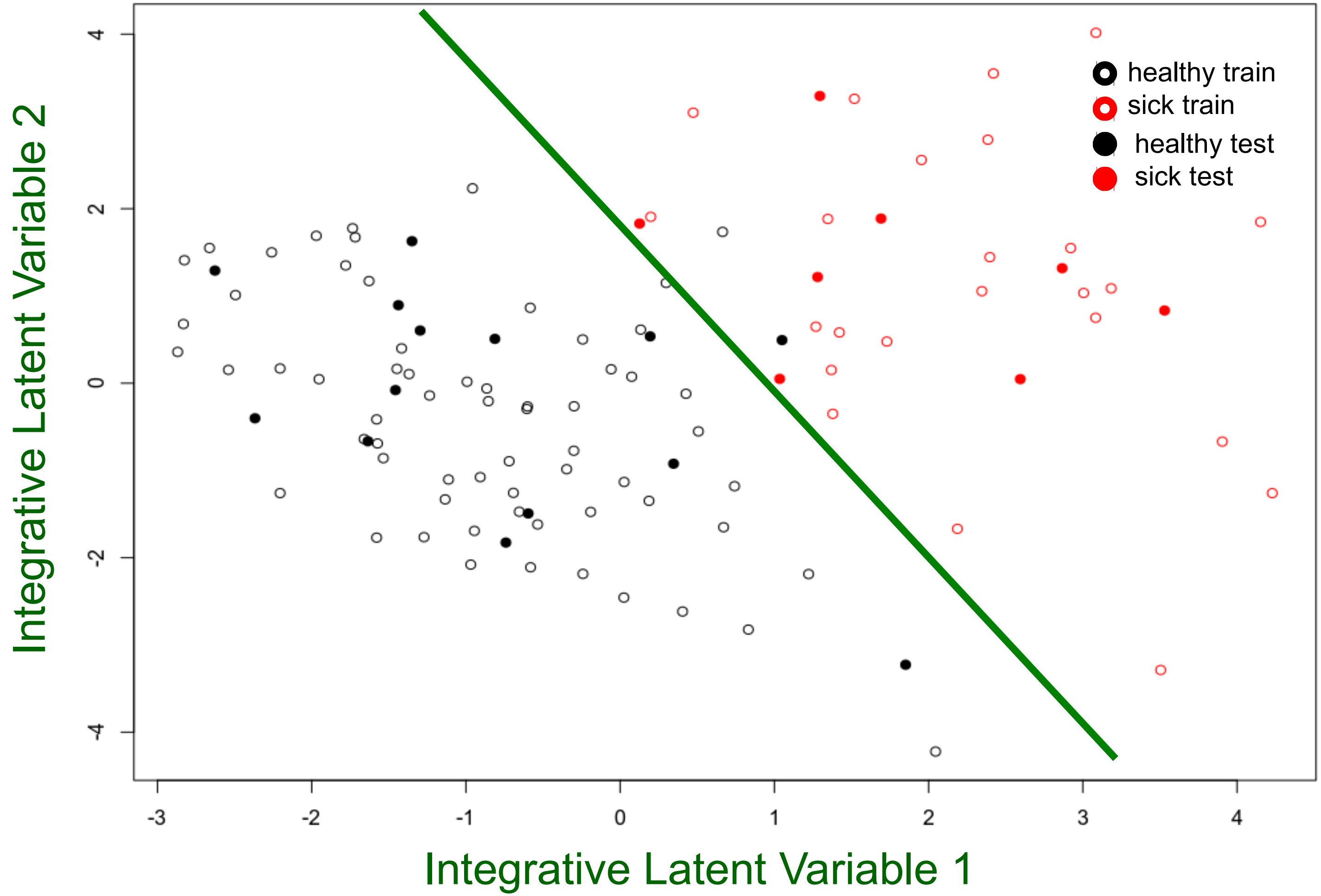


Feature Selection

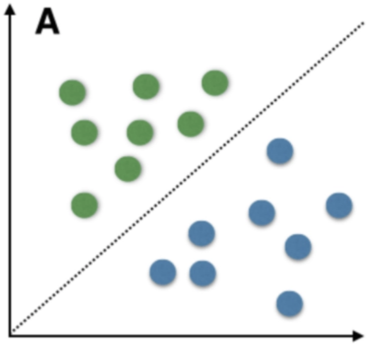
Omics Integration

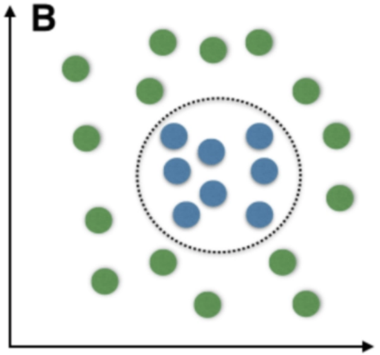
Trained Model

Linear Separation: Decision Boundary



A: Linearly Separable Data

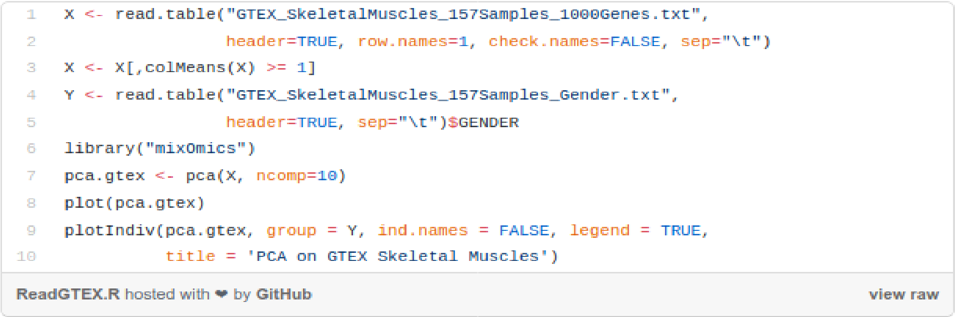




B: Non-Linearly Separable Data

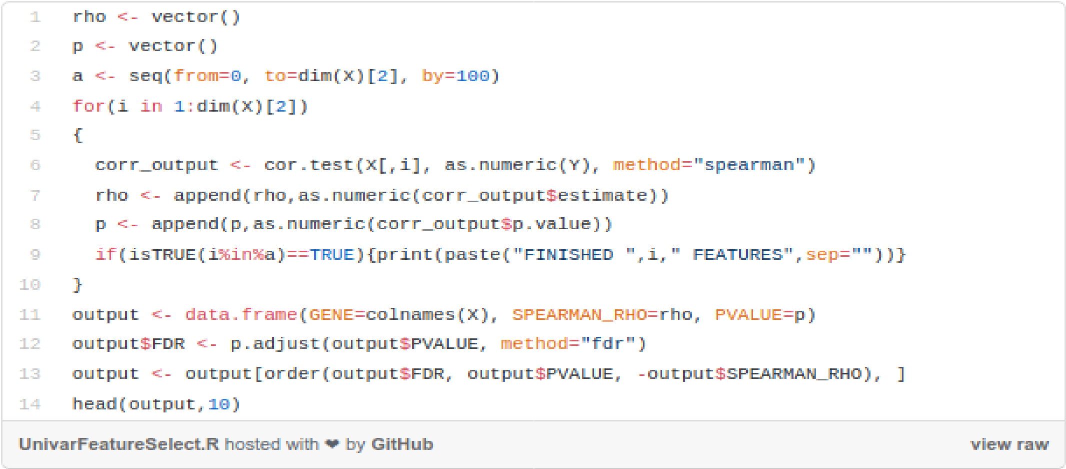
Univariate and Multivariate Feature Selection

Univariate Feature Selection



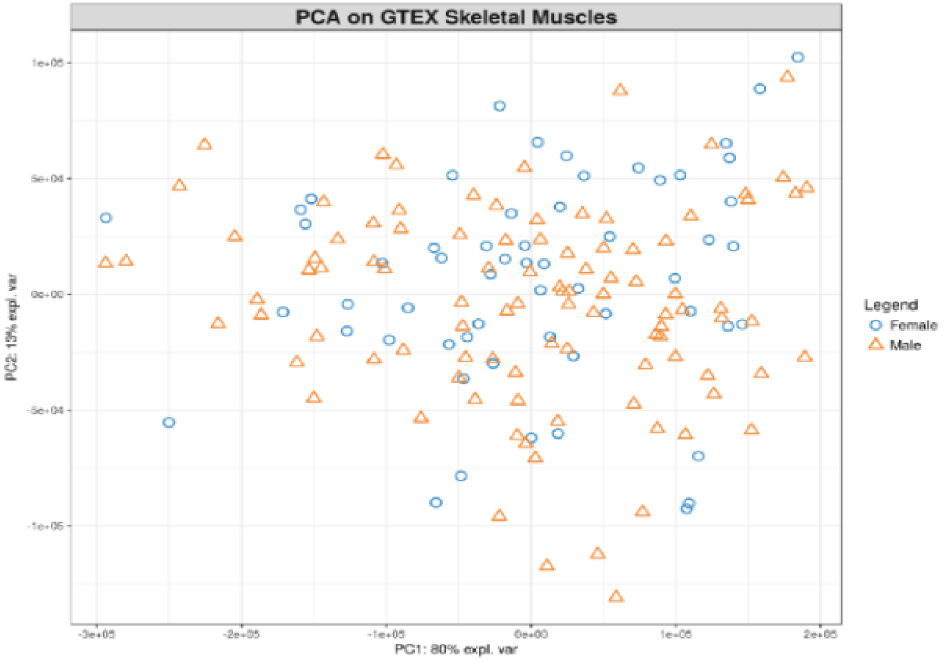
```
X <- read.table("GTEX_SkeletalMuscles_157Samples_1000Genes.txt",
                header=TRUE, row.names=1, check.names=FALSE, sep="\t")
X <- X[,colMeans(X) >= 1]
Y <- read.table("GTEX_SkeletalMuscles_157Samples_Gender.txt",
                header=TRUE, sep="\t")$GENDER
library("mixOmics")
pca.gtex <- pca(X, ncomp=10)
plot(pca.gtex)
plotIndiv(pca.gtex, group = Y, ind.names = FALSE, legend = TRUE,
          title = 'PCA on GTEX Skeletal Muscles')
```

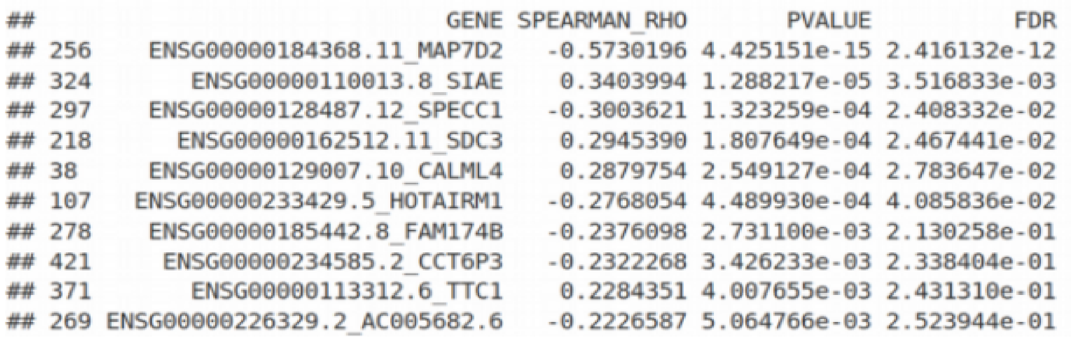
ReadGTEX.R hosted with ❤ by GitHub   view raw



```
rho <- vector()
p <- vector()
a <- seq(from=0, to=dim(X)[2], by=100)
for(i in 1:dim(X)[2])
{
  corr_output <- cor.test(X[,i], as.numeric(Y), method="spearman")
  rho <- append(rho,as.numeric(corr_output$estimate))
  p <- append(p,as.numeric(corr_output$p.value))
  if(isTRUE(i%in%a)==TRUE){print(paste("FINISHED ",i," FEATURES",sep=""))}
}
output <- data.frame(GENE=colnames(X), SPEARMAN_RHO=rho, PVALUE=p)
output$FDR <- p.adjust(output$PVALUE, method="fdr")
output <- output[order(output$FDR, output$PVALUE, -output$SPEARMAN_RHO), ]
head(output,10)
```

UnivarFeatureSelect.R hosted with ❤ by GitHub   view raw





```
##                             GENE SPEARMAN_RHO       PVALUE          FDR
## 256     ENSG00000184368.11_MAP7D2   -0.5730196 4.425151e-15 2.416132e-12
## 324       ENSG00000110013.8_SIAE    0.3403994 1.288217e-05 3.516833e-03
## 297     ENSG00000128487.12_SPECC1   -0.3003621 1.323259e-04 2.408332e-02
## 218       ENSG00000162512.11_SDC3    0.2945390 1.807649e-04 2.467441e-02
## 38      ENSG00000129007.10_CALML4    0.2879754 2.549127e-04 2.783647e-02
## 107    ENSG00000233429.5_HOTAIRM1   -0.2768054 4.489930e-04 4.085836e-02
## 278     ENSG00000185442.8_FAM174B   -0.2376098 2.731100e-03 2.130258e-01
## 421      ENSG00000234585.2_CCT6P3   -0.2322268 3.426233e-03 2.338404e-01
## 371        ENSG00000113312.6_TTC1    0.2284351 4.007655e-03 2.431310e-01
## 269 ENSG00000226329.2_AC005682.6   -0.2226587 5.064766e-03 2.523944e-01
```

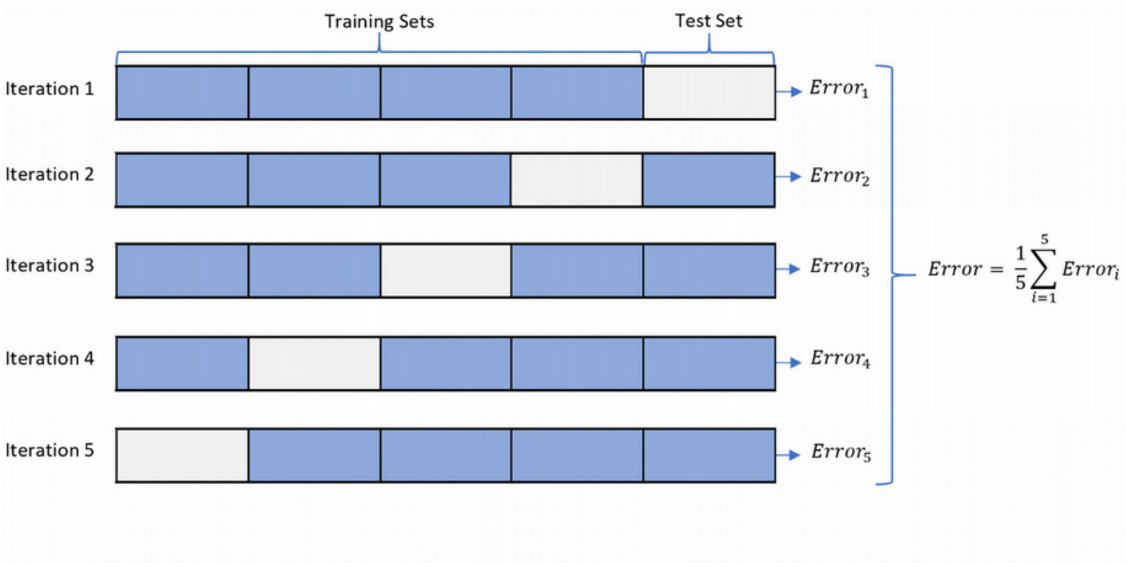
Generally acknowledged that univariate feature selection has poor predictive capacity compared to multivariate feature selection

Multivariate Feature Selection: LASSO

$$Y = \beta_1 X_1 + \beta_2 X_2 + \epsilon$$

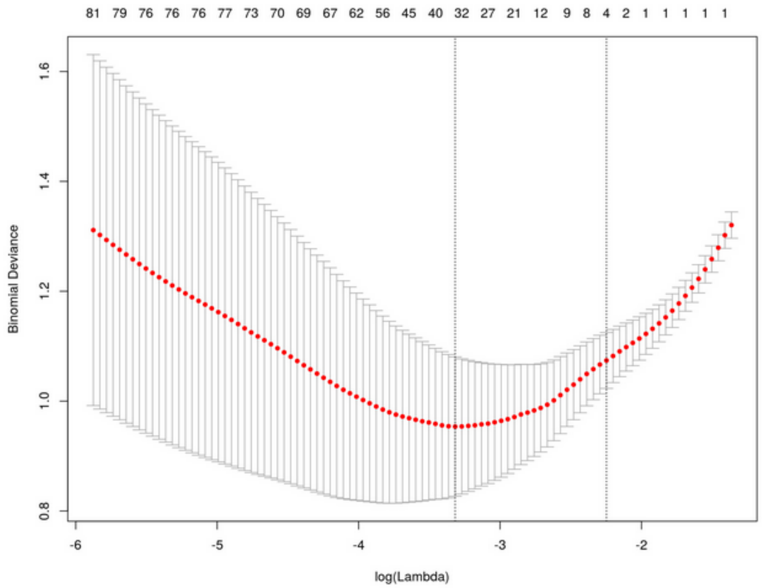
$$\text{OLS} = (Y - \beta_1 X_1 - \beta_2 X_2)^2$$

$$\text{Penalized OLS} = (Y - \beta_1 X_1 - \beta_2 X_2)^2 + \lambda(|\beta_1| + |\beta_2|)$$



$$Error = \frac{1}{5}\sum_{i=1}^{5} Error_i$$

Cross-validation is a standard way to tune model hyperparameters such as λ in LASSO



Regularizations are Priors in Bayesian stats

$$Y = \beta_1 X_1 + \beta_2 X_2 + \epsilon; \quad Y \sim N(\,\beta_1 X_1 + \beta_2 X_2, \sigma^2\,) \equiv \mathrm{L}\,(\,\mathrm{Y}\,|\,\beta_1, \beta_2\,)$$

- **Maximum Likelihood** principle: maximize probability to observe data given parameters:

$$\mathrm{L}\,(\,\mathrm{Y}\,|\,\beta_1, \beta_2\,) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{-\dfrac{(\mathrm{Y} - \beta_1 \mathrm{X}_1 - \beta_2 \mathrm{X}_2)^2}{2\sigma^2}}$$

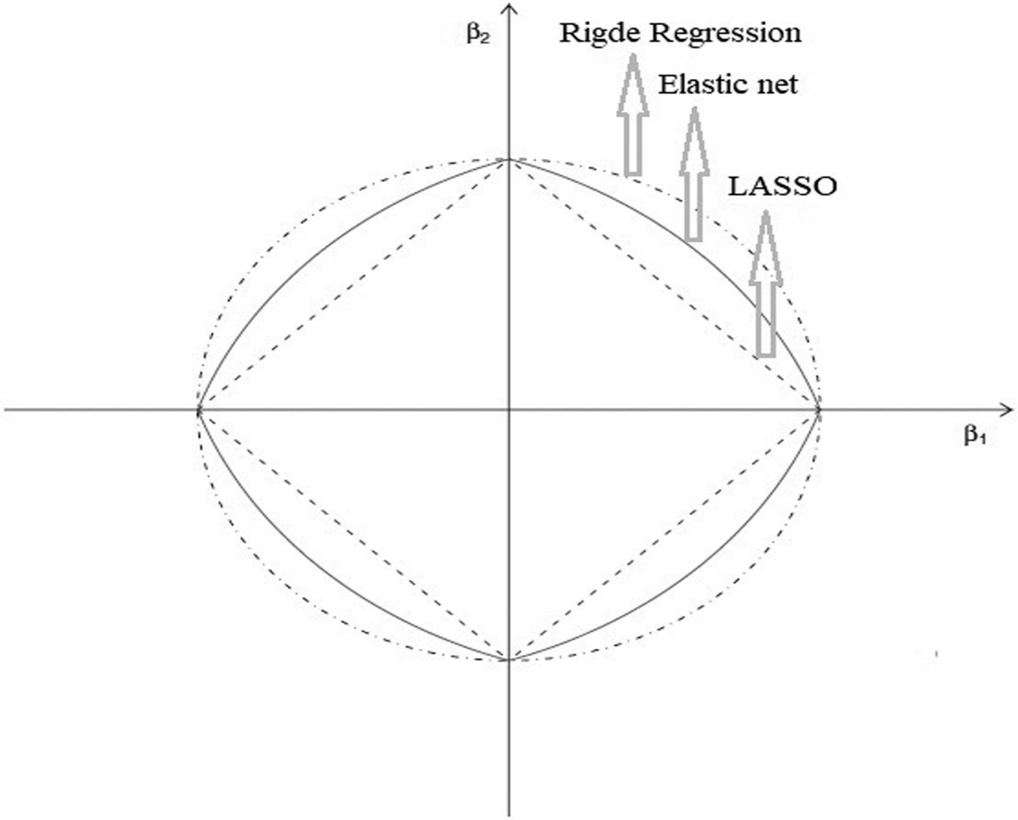
- **Bayes theorem**: maximize posterior probability of observing parameters given data:

$$\mathrm{Posterior}(\mathrm{params}\,|\,\mathrm{data}) = \frac{\mathrm{L}(\mathrm{data}\,|\,\mathrm{params}) * \mathrm{Prior}(\mathrm{params})}{\int \mathrm{L}(\mathrm{data}\,|\,\mathrm{params}) * \mathrm{Prior}(\mathrm{params})\,\mathrm{d}(\mathrm{params})}$$

$$\mathrm{Posterior}(\,\beta_1, \beta_2\,|\,\mathrm{Y}\,) \sim \mathrm{L}\,(\,\mathrm{Y}\,|\,\beta_1, \beta_2\,) * \mathrm{Prior}(\beta_1, \beta_2) \sim \exp^{-\frac{(\mathrm{Y} - \beta_1 \mathrm{X}_1 - \beta_2 \mathrm{X}_2)^2}{2\sigma^2}} * \exp^{-\lambda(|\beta_1| + |\beta_2|)}$$

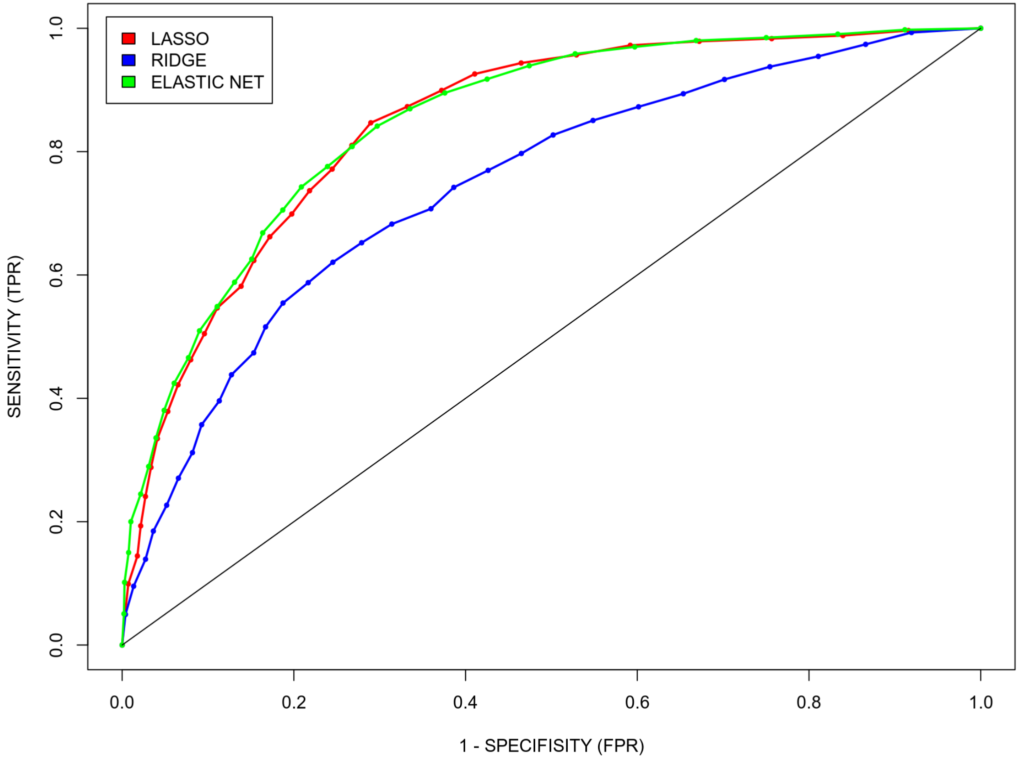
$$-\log\left[\mathrm{Posterior}(\,\beta_1, \beta_2\,|\,\mathrm{Y}\,)\right] \sim (\mathrm{Y} - \beta_1 \mathrm{X}_1 - \beta_2 \mathrm{X}_2)^2 + \lambda(|\beta_1| + |\beta_2|)$$

Lasso vs. Ridge vs. Elastic Net



$$\text{Lasso}: |\beta_1| + |\beta_2| \le \lambda$$

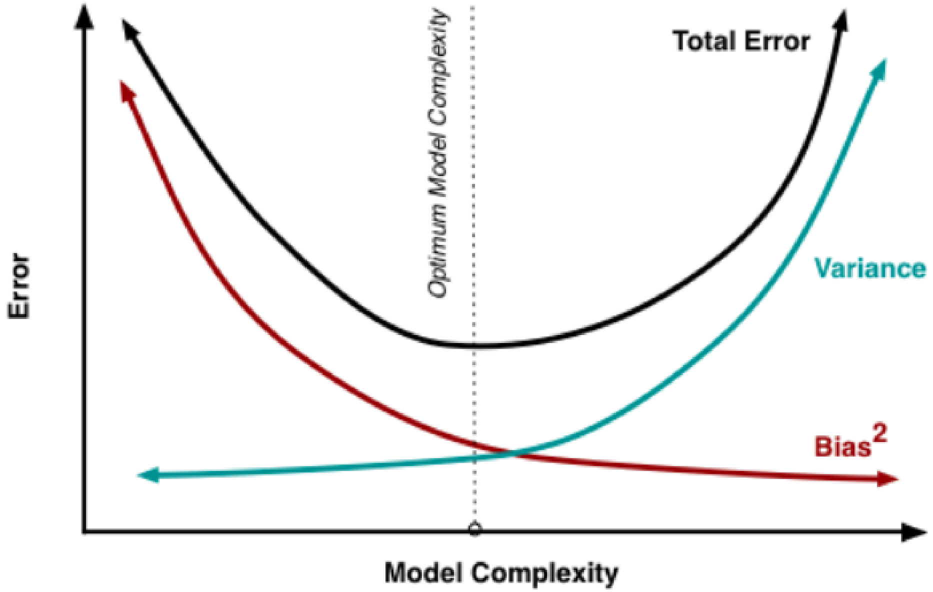
$$\text{Ridge}: \beta_1^2 + \beta_2^2 \le \lambda$$

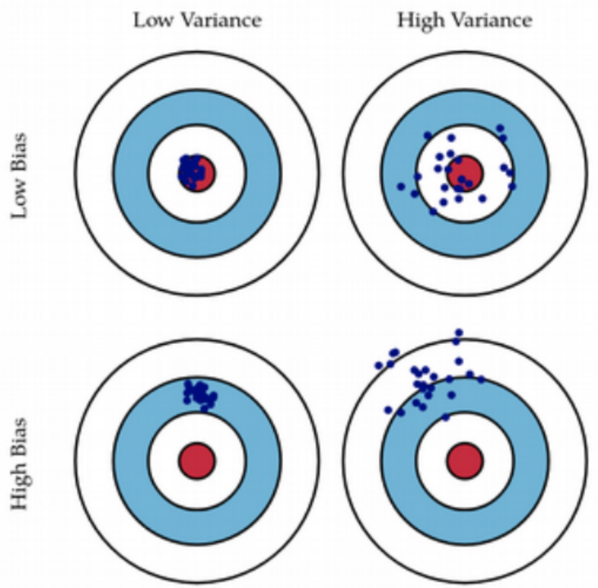


Lasso is more conservative

Ridge is more permissive

Penalized regression interpretation

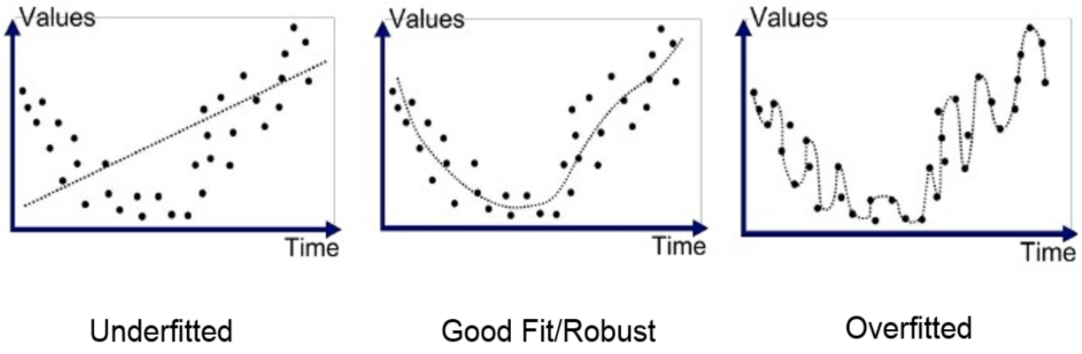




$$Y = f(X) \Longrightarrow \text{Reality}$$

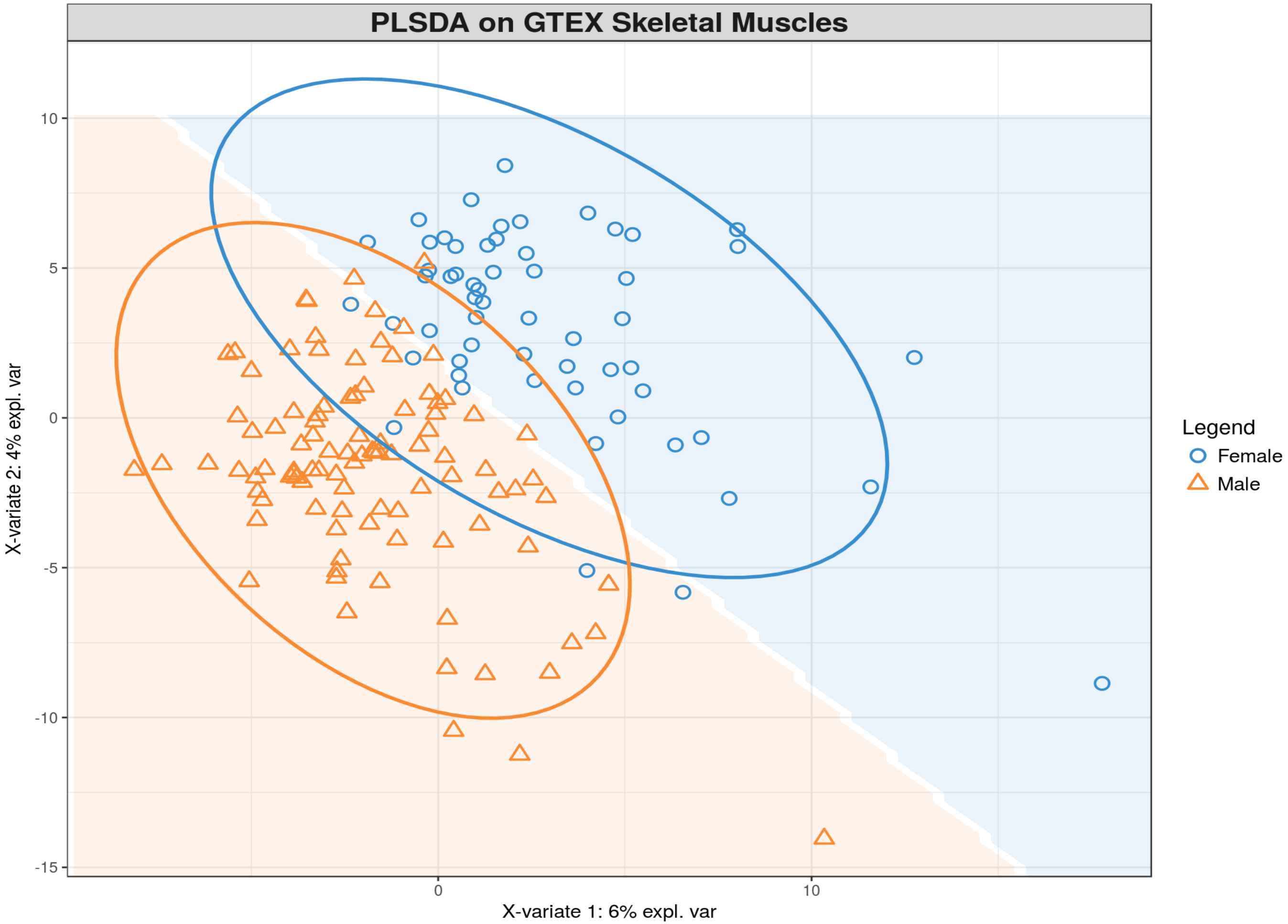
$$\mathrm{Y} = \hat{\mathrm{f}}(\mathrm{X}) + \mathrm{Error} \Longrightarrow \mathrm{Model}$$

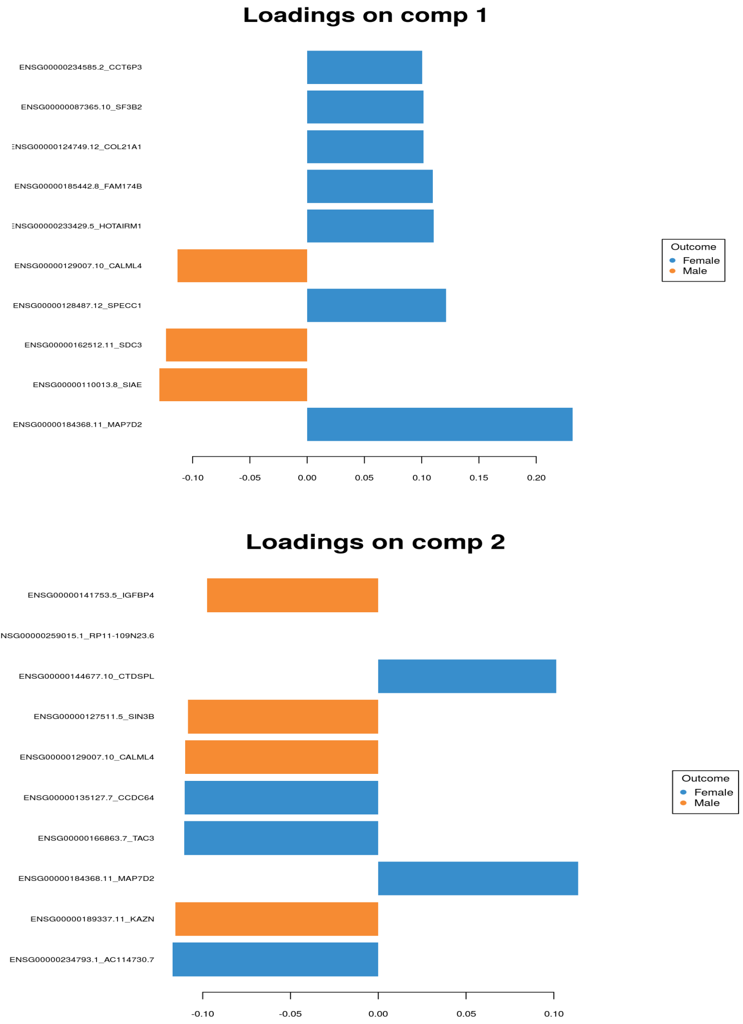
$$\mathrm{Error}^2 = (\mathrm{Y} - \hat{\mathrm{f}}(\mathrm{X}))^2 = \mathrm{Bias}^2 + \mathrm{Variance}$$



LASSO – high bias, low variance

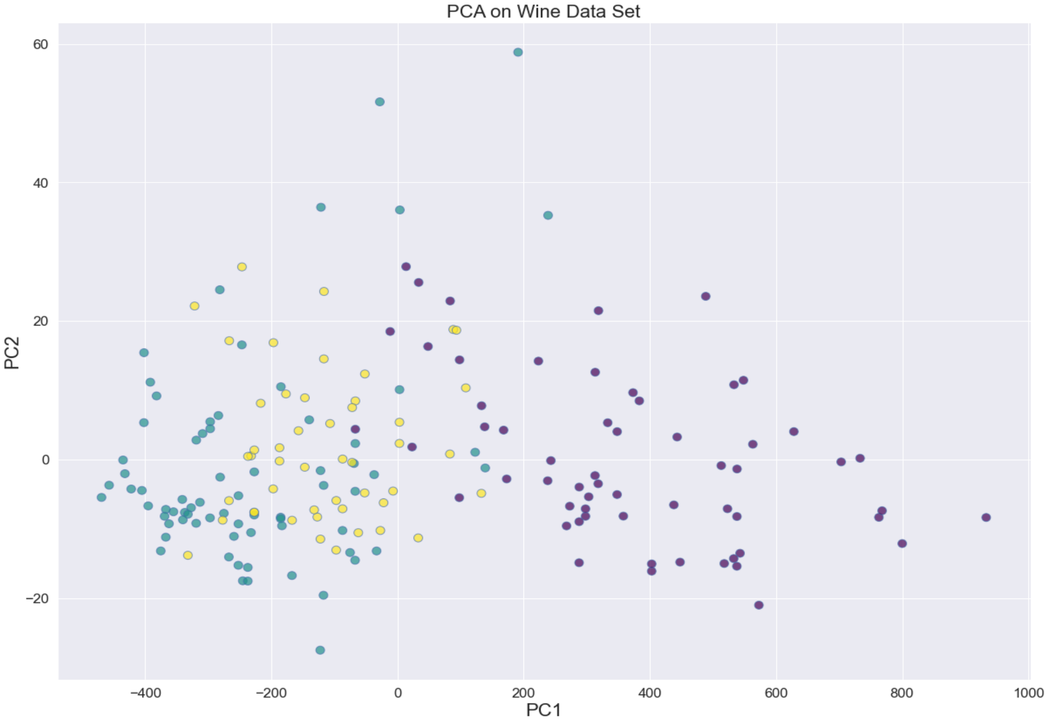
Multivariate Feature Selection: PLS

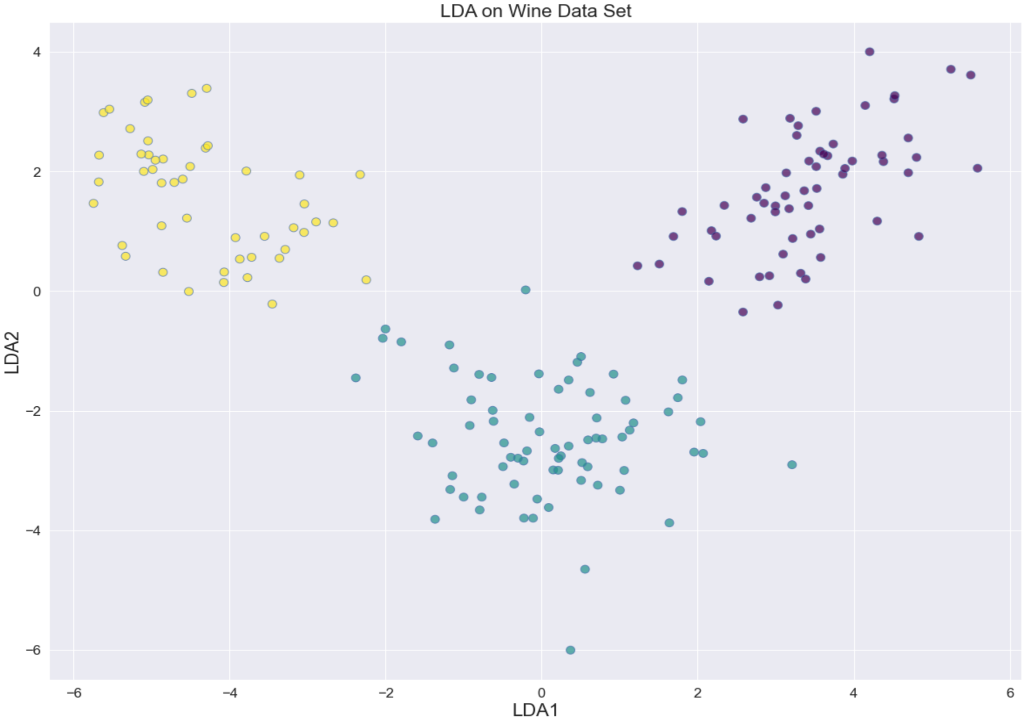




Select features that separate two groups of samples the most

Multivariate Feature Selection: Linear Discriminant Analysis (LDA)

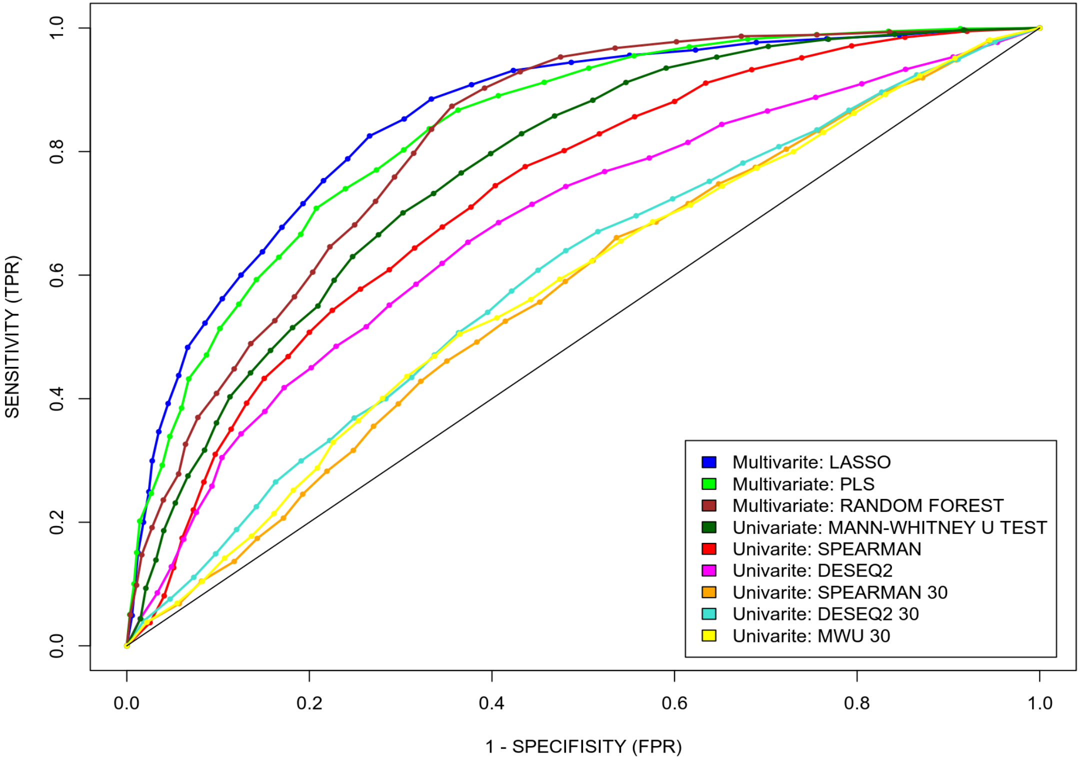




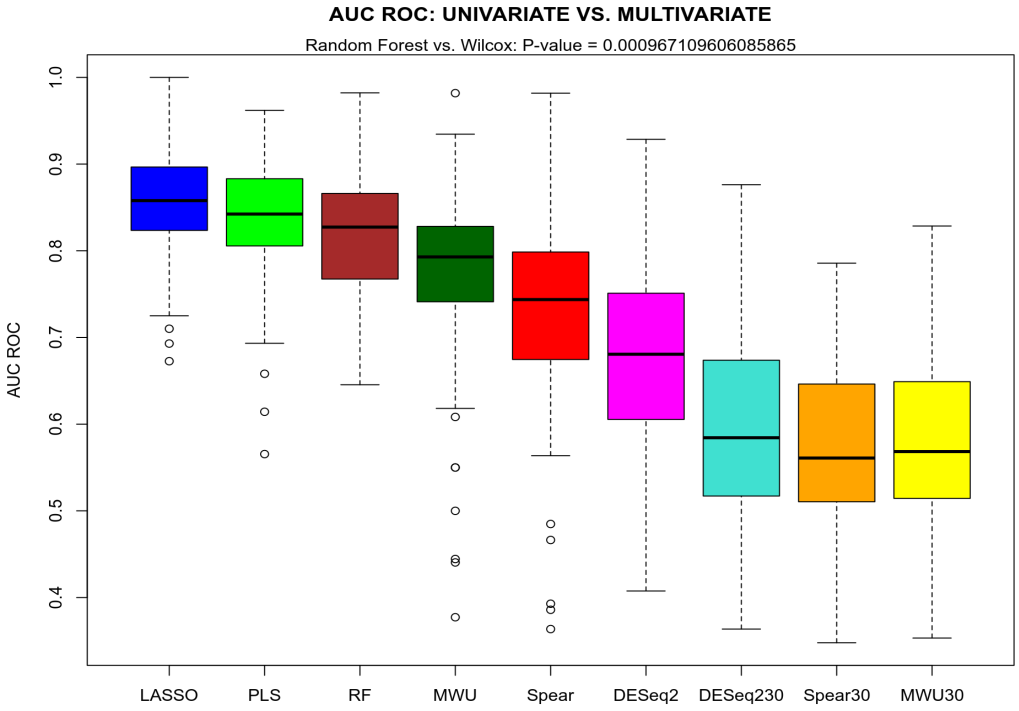
Minimize variance within clusters and maximize variance between clusters

Similar to what ANOVA is doing, therefore LINEAR Discriminant Analysis (LDA)

Univariate vs. Multivariate Prediction



Multivariate methods (LASSO, PLS, RanFor) have significantly higher AUC ROC than univariate methods (Spear, MWU, DESeq2) on skeletal muscle gene expression data

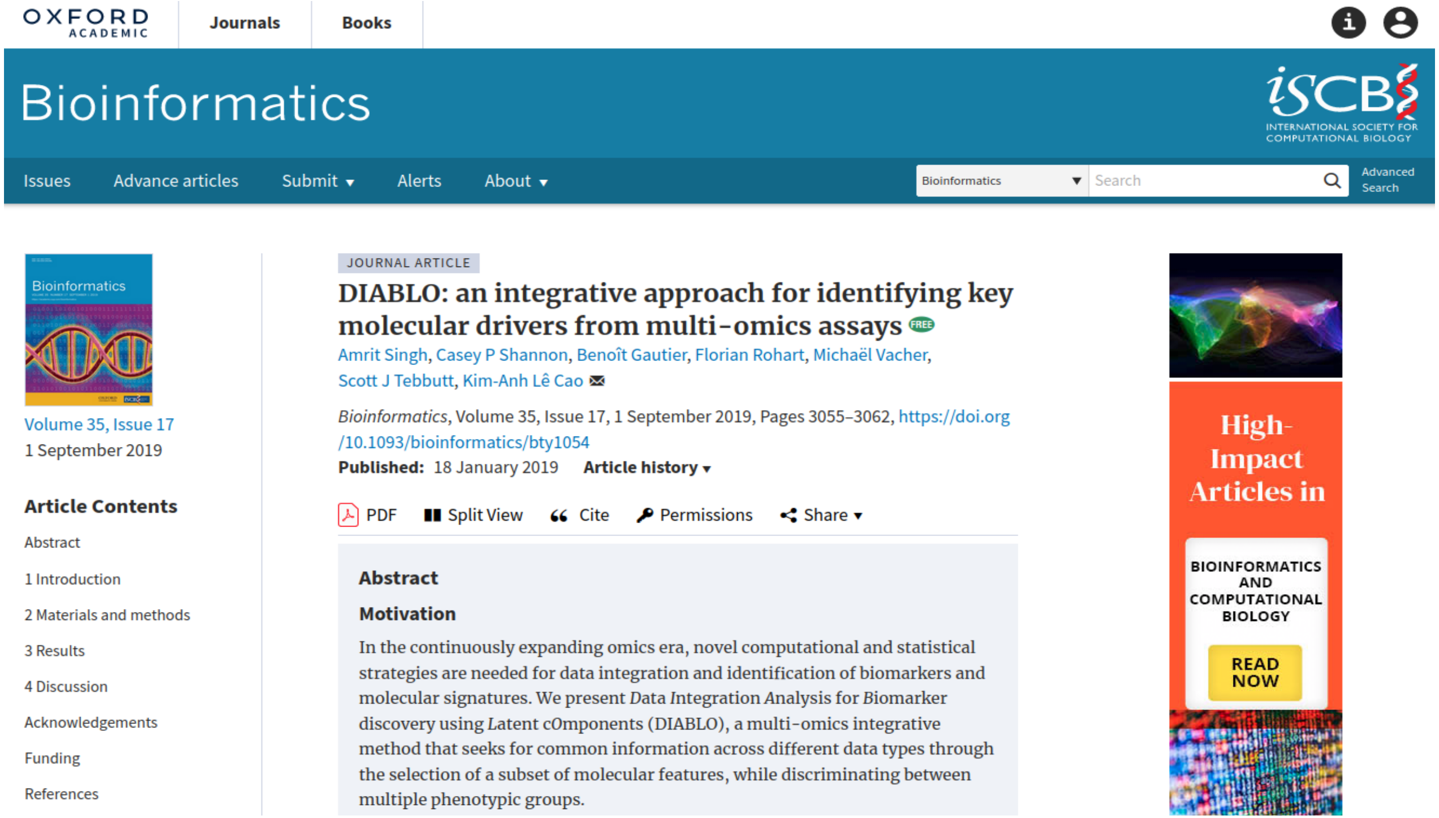


AUC ROC: UNIVARIATE VS. MULTIVARIATE

Random Forest vs. Wilcox: P-value = 0.000967109606085865

If you find a dataset where univariate feature selection has higher predictive capacity than multivariate one, please let me know

DIABLO Omics Integration



Volume 35, Issue 17
1 September 2019

JOURNAL ARTICLE

DIABLO: an integrative approach for identifying key molecular drivers from multi-omics assays FREE

Amrit Singh, Casey P Shannon, Benoît Gautier, Florian Rohart, Michaël Vacher, Scott J Tebbutt, Kim-Anh Lê Cao ✉

Bioinformatics, Volume 35, Issue 17, 1 September 2019, Pages 3055–3062, https://doi.org/10.1093/bioinformatics/bty1054

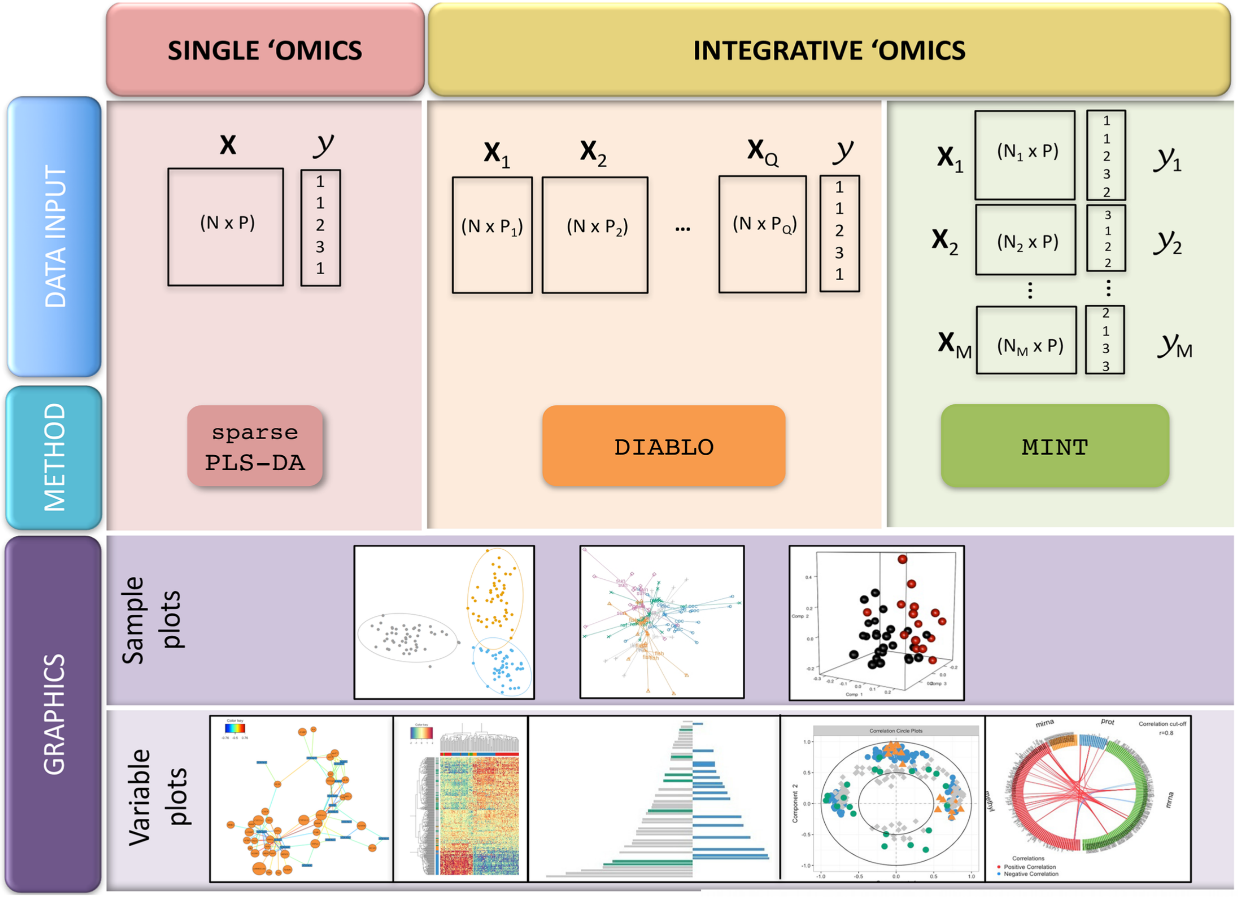
Published: 18 January 2019 **Article history** ▾

Abstract

Motivation

In the continuously expanding omics era, novel computational and statistical strategies are needed for data integration and identification of biomarkers and molecular signatures. We present Data Integration Analysis for Biomarker discovery using Latent cOmponents (DIABLO), a multi-omics integrative method that seeks for common information across different data types through the selection of a subset of molecular features, while discriminating between multiple phenotypic groups.

DIABLO PLS-based algorithm



SINGLE 'OMICS

INTEGRATIVE 'OMICS

DATA INPUT

$\mathbf{X}$ (N x P) y 1 1 2 3 1

$\mathbf{X}_1$ (N x P$_1$) $\mathbf{X}_2$ (N x P$_2$) … $\mathbf{X}_Q$ (N x P$_Q$) y 1 1 2 3 1

$\mathbf{X}_1$ (N$_1$ x P) 1 1 2 3 2 y_1

$\mathbf{X}_2$ (N$_2$ x P) 3 1 2 2 y_2

$\mathbf{X}_M$ (N$_M$ x P) 2 1 3 3 y_M

METHOD

sparse PLS-DA

DIABLO

MINT

GRAPHICS

Sample plots

Variable plots

Denote Q normalized, centered and scaled datasets $X^{(1)}$ $(N \times P_1)$, $X^{(2)}(N \times P_2)$,..., $X^{(Q)}(N \times P_Q)$ measuring the expression levels of $P_1, \ldots, P_Q$ 'omics variables on the same N samples'. sGCCA solves the optimization function for each dimension $h = 1, \ldots, H$:

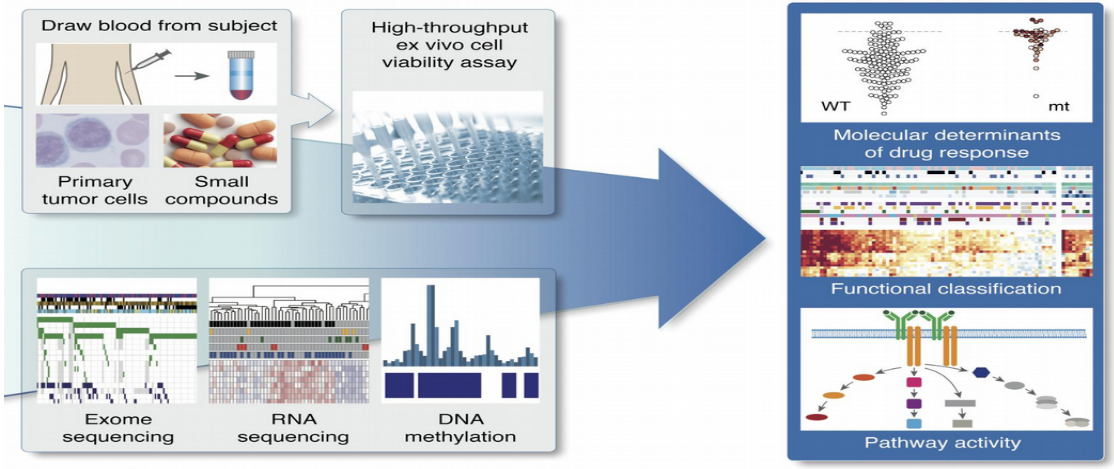
$$\max_{a_h^{(1)},\ldots,a_h^{(Q)}} \sum_{i,j=1,i\neq j}^{Q} c_{i,j} \ \mathrm{cov}(X_h^{(i)}a_h^{(i)}, X_h^{(j)}a_h^{(j)}), \qquad (1)$$

$$\text{s.t. } ||a_h^{(q)}||_2 = 1 \text{ and } ||a_h^{(q)}||_1 \leq \lambda^{(q)} \textit{ for all } 1 \leq q \leq Q$$

where $a_h^{(q)}$ is the variable coefficient or loading vector on dimension h associated to the residual matrix $X_h^{(q)}$ of the dataset $X^{(q)}$. $C = \{c_{i,j}\}_{i,j}$ is a $(Q \times Q)$ design matrix that specifies whether datasets should be connected. Elements in C can be set to zeros when datasets are not connected and ones where datasets are fully connected, as we further describe in Section 2.2. In addition in (1), $\lambda^{(q)}$ is a non-negative parameter that controls the amount of shrinkage and thus the number of non-zero coefficients in $a_h^{(q)}$. Similar to the LASSO (Tibshirani, 1996) and other ℓ_1 penalized multivariate models developed for single omics analysis (Lê Cao *et al.*, 2011), the penalization enables the selection of a subset of variables with non-zero coefficients that define each component score $t_h^{(q)} = X_h^{(q)} a_h^{(q)}$. The result is the identification of variables that are highly correlated *between* and *within* omics datasets.

$$\max_{\beta} \mathrm{cov}(X, Y) \Longrightarrow \hat{\beta}$$

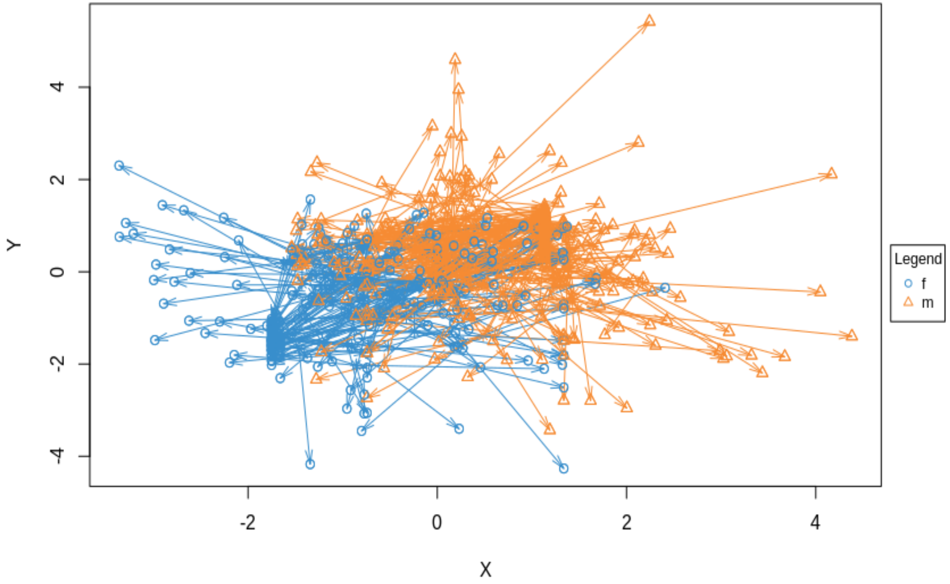
DIABLO Omics Integration

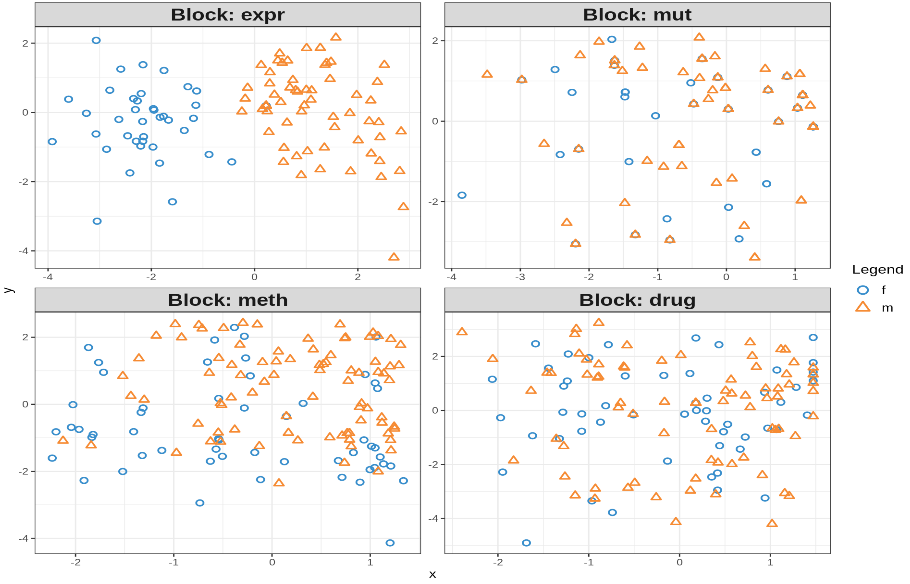


Chronic Lymphocytic Leukaemia (CLL):

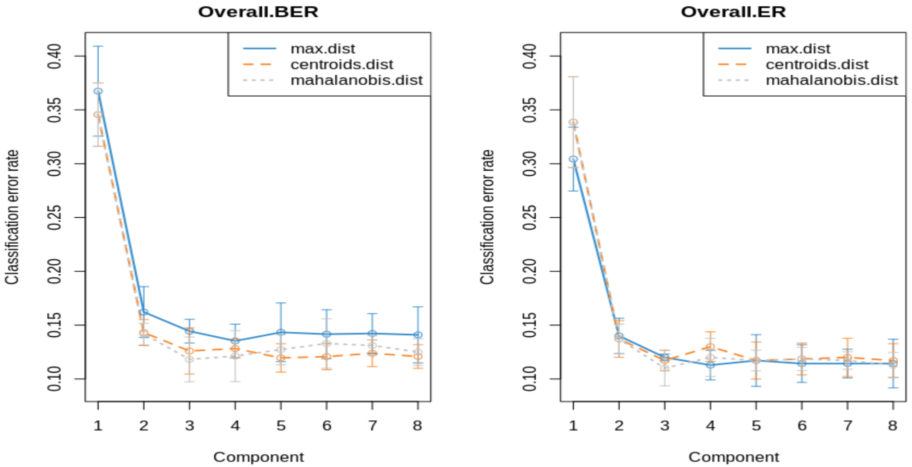
gene expression (RNAseq), mutations, methylation, drug response

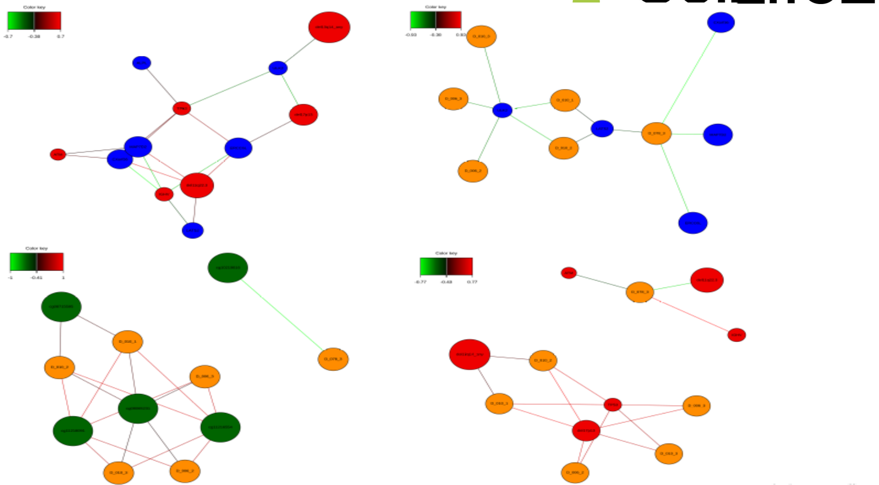
Dietrich et al., J Clin Invest. 2018



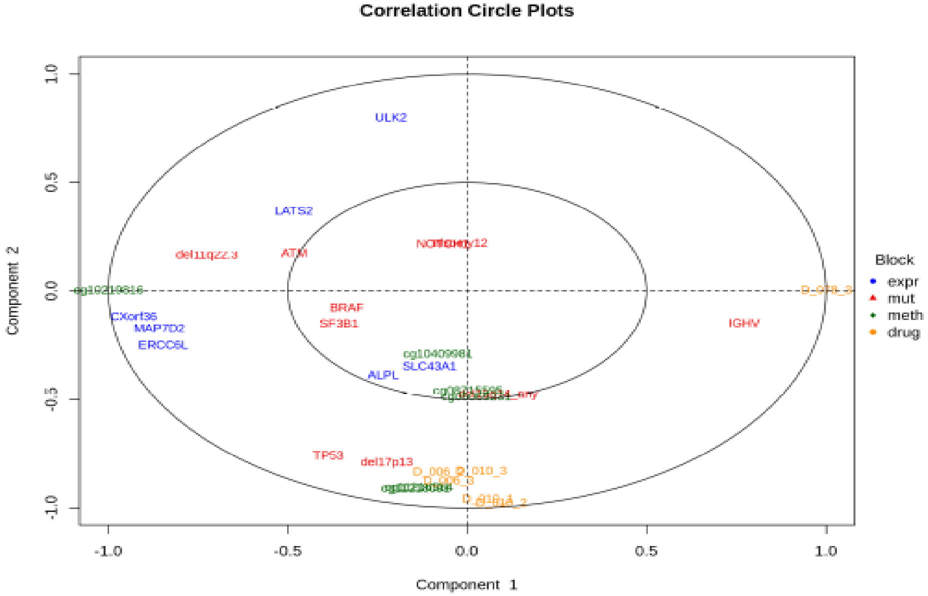


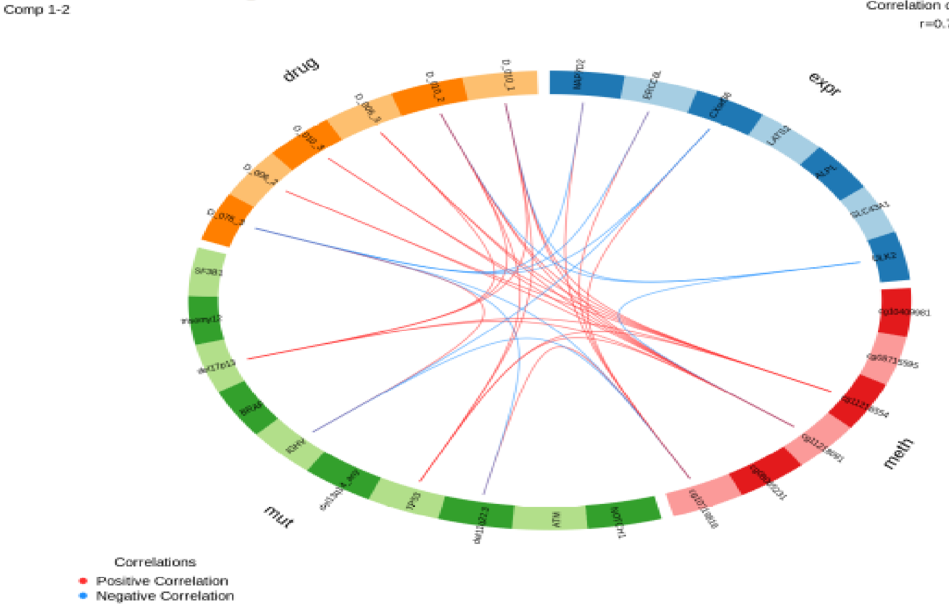
DIABLO visualization



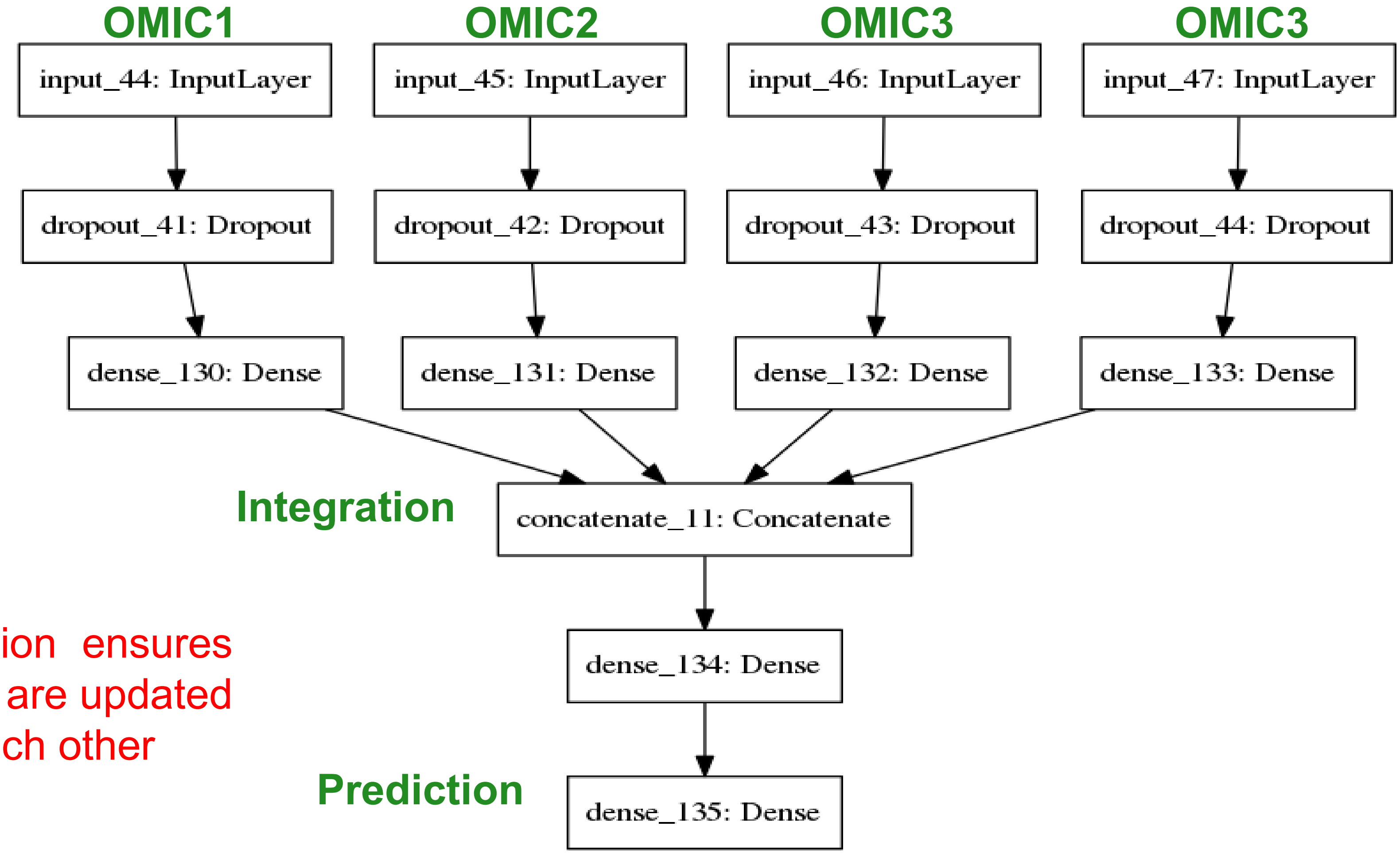


Correlation cut-off
r=0.7





Supervised deep learning Omics integration



Regularization

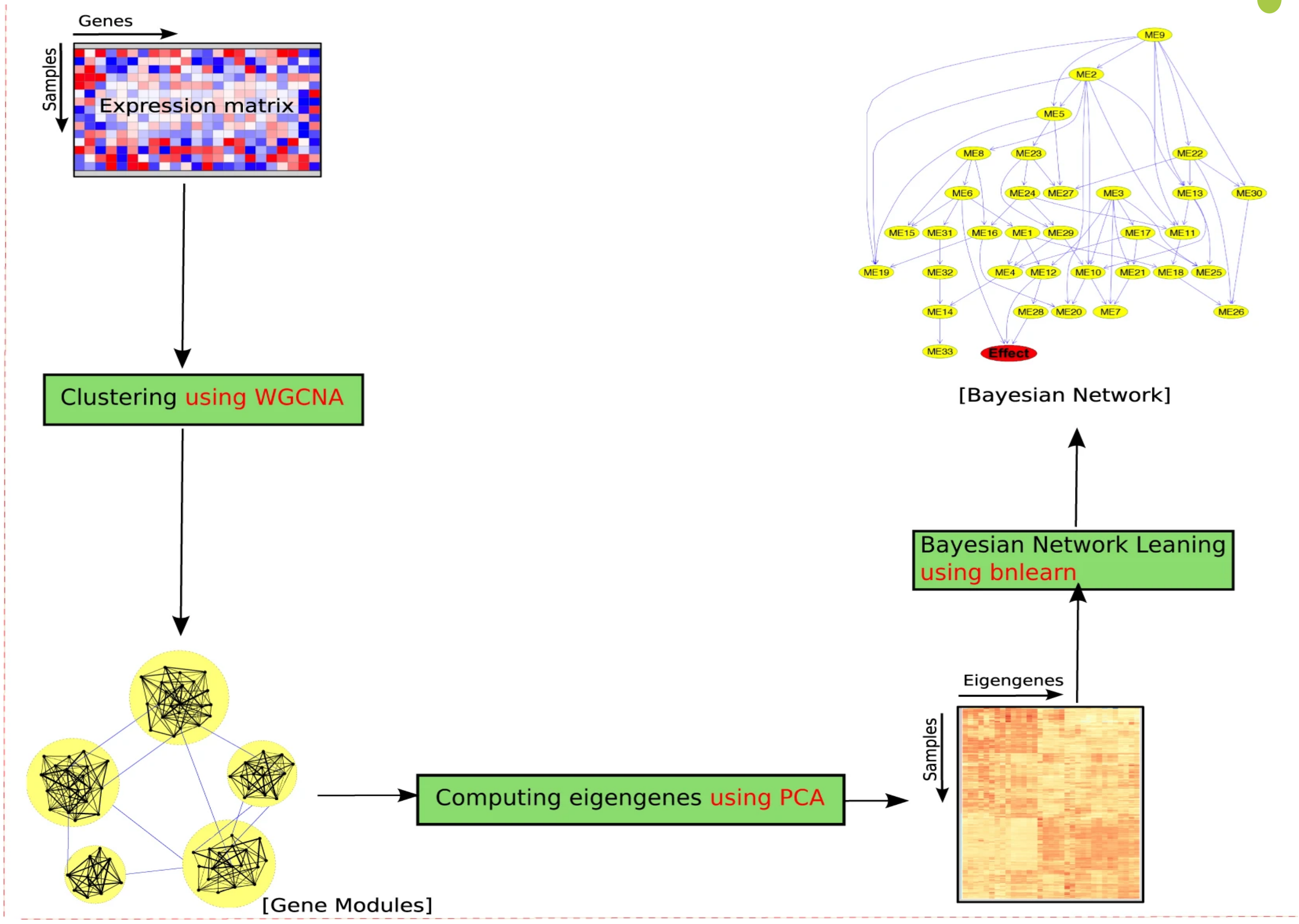
Conversion to nonparametric space

Integration

Prediction

Back propagation ensures that all weights are updated in context of each other

Bayesian Networks for Omics Integration



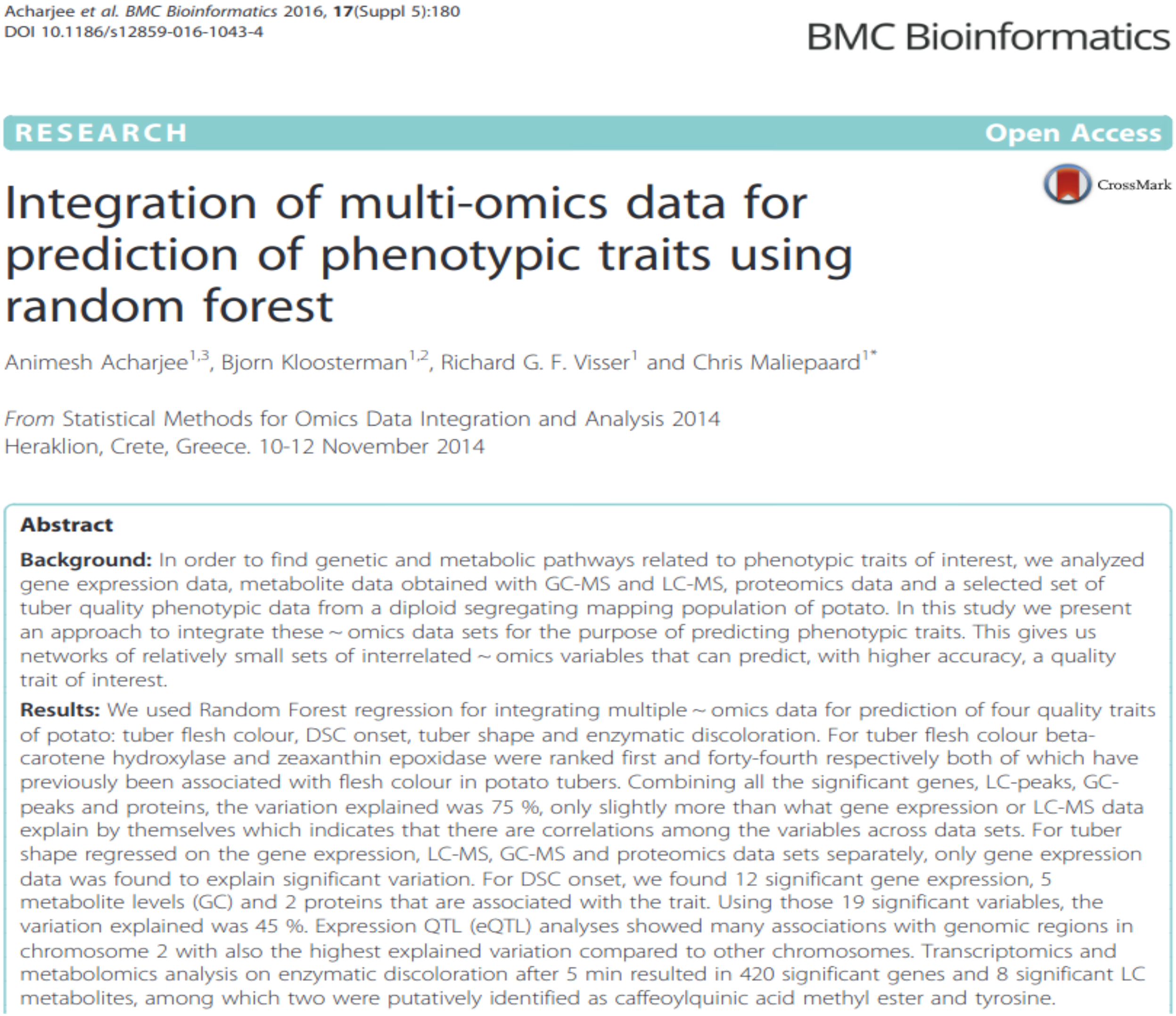
Clustering using WGCNA

Computing eigengenes using PCA

Bayesian Network Leaning using bnlearn

Agrahari R. et al., Sci Rep. 2018 3;8(1):6951

Random Forest for Omics Integration



Acharjee *et al. BMC Bioinformatics* 2016, **17**(Suppl 5):180
DOI 10.1186/s12859-016-1043-4

BMC Bioinformatics

RESEARCH Open Access

Integration of multi-omics data for prediction of phenotypic traits using random forest

Animesh Acharjee[1,3], Bjorn Kloosterman[1,2], Richard G. F. Visser[1] and Chris Maliepaard[1*]

From Statistical Methods for Omics Data Integration and Analysis 2014
Heraklion, Crete, Greece. 10-12 November 2014

Abstract

Background: In order to find genetic and metabolic pathways related to phenotypic traits of interest, we analyzed gene expression data, metabolite data obtained with GC-MS and LC-MS, proteomics data and a selected set of tuber quality phenotypic data from a diploid segregating mapping population of potato. In this study we present an approach to integrate these ~ omics data sets for the purpose of predicting phenotypic traits. This gives us networks of relatively small sets of interrelated ~ omics variables that can predict, with higher accuracy, a quality trait of interest.

Results: We used Random Forest regression for integrating multiple ~ omics data for prediction of four quality traits of potato: tuber flesh colour, DSC onset, tuber shape and enzymatic discoloration. For tuber flesh colour beta-carotene hydroxylase and zeaxanthin epoxidase were ranked first and forty-fourth respectively both of which have previously been associated with flesh colour in potato tubers. Combining all the significant genes, LC-peaks, GC-peaks and proteins, the variation explained was 75 %, only slightly more than what gene expression or LC-MS data explain by themselves which indicates that there are correlations among the variables across data sets. For tuber shape regressed on the gene expression, LC-MS, GC-MS and proteomics data sets separately, only gene expression data was found to explain significant variation. For DSC onset, we found 12 significant gene expression, 5 metabolite levels (GC) and 2 proteins that are associated with the trait. Using those 19 significant variables, the variation explained was 45 %. Expression QTL (eQTL) analyses showed many associations with genomic regions in chromosome 2 with also the highest explained variation compared to other chromosomes. Transcriptomics and metabolomics analysis on enzymatic discoloration after 5 min resulted in 420 significant genes and 8 significant LC metabolites, among which two were putatively identified as caffeoylquinic acid methyl ester and tyrosine.

NBIS

National Bioinformatics Infrastructure Sweden (NBIS)

SciLifeLab

Knut och Alice Wallenbergs Stiftelse

Vetenskapsrådet

LUNDS UNIVERSITET