How To Save An Awful-Designed Project?

… with the recourse of machine learning approaches for multi-omics analysis

Lijiao NING (lijiao.ning@cnrs.fr)

Inserm U1283 / CNRS UMR8199 / Université de Lille

6th October 2022

RIS Thematic Day, Paris

There is a project …

A project with ...

👍 Variety of data

- >300 samples
- >30 phenotypes
- different omics data
  - genotype
  - methylation
  - RNA-seq

Abbreviations

- *NALF*: non-alcoholic fatty liver
- *NASH*: non-alcoholic steatohepatitis
- *BMI*: body mass index
- *IGT*: impaired glucose tolerance
- *T2D*: type 2 diabetes

But confounding factors 😩

Characteristic	**Control**, N = 80[1]	**NAFL**, N = 137[1]	**NASH**, N = 83[1]	**p-value**[2]
Age	34 (26, 43)	43 (32, 51)	46 (40, 56)	**<0.001**
Sex				**0.009**
Male	13 (16%)	48 (35%)	28 (34%)	
Female	67 (84%)	89 (65%)	55 (66%)	
BMI	43 (40, 49)	45 (42, 51)	44 (41, 51)	0.2
Diabetic Status				**<0.001**
Normoglycemic	42 (57%)	17 (12%)	5 (6.1%)	
IGT	16 (22%)	54 (40%)	11 (13%)	
T2D	16 (22%)	65 (48%)	66 (80%)	
Unknown	6	1	1	

[1] Median (IQR); n (%)
[2] Kruskal-Wallis rank sum test; Pearson's Chi-squared test

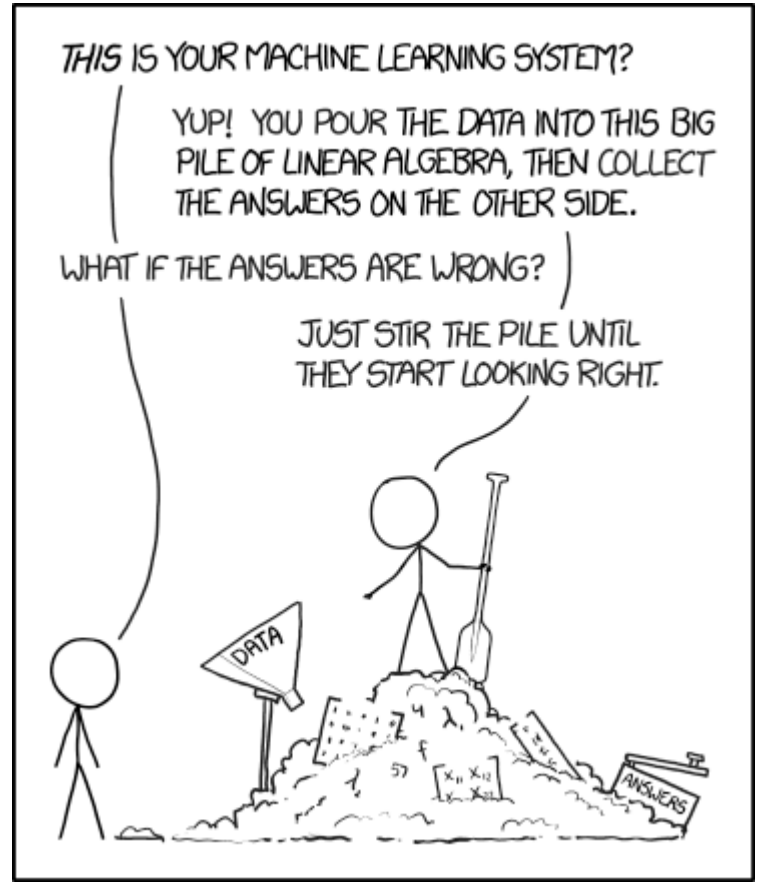
What can I do?

- ~~Abandon this project and the next please! 😉~~

- Select a subset of matched sample and follow the original design
  - Keep all groups → 0 matched
  - Only NAFL and NASH groups → 166 match samples

- Use all available data with machine learning approaches
  - Find phenotype-based clusters
  - Identify multi-omic features related to the clusters

Let’s start!


THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS

Phenotype-based clustering

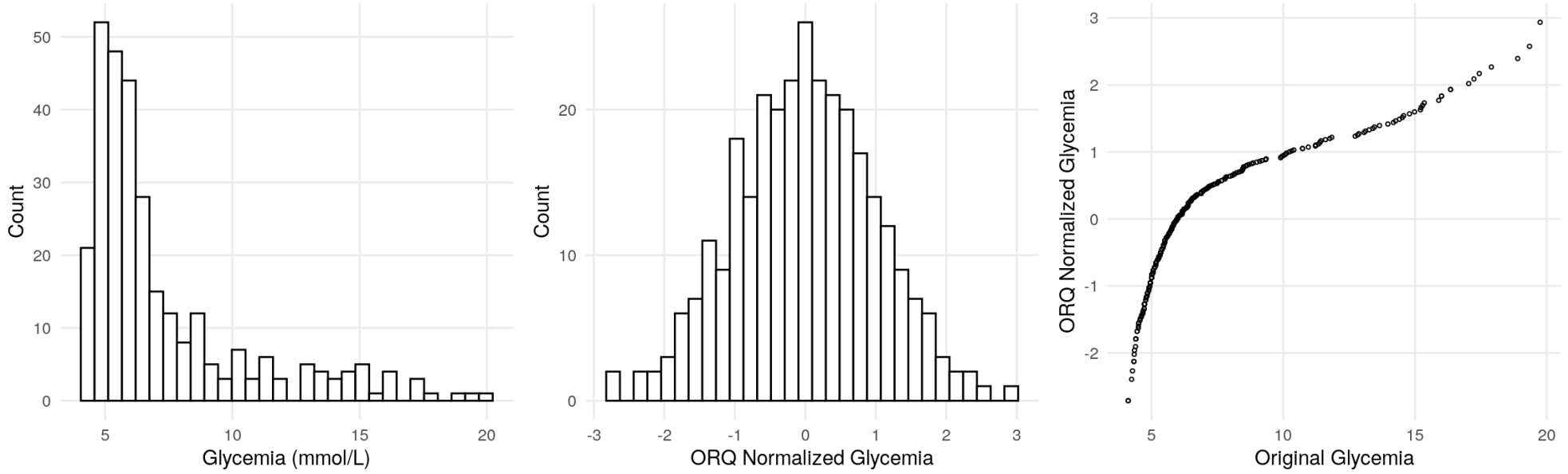
Available phenotypes

- Age, sex, BMI
- Liver biopsy
  - Scores for steatosis, lobular inflammation, ballooning
  - Brunt score, fibrosis score, NAS (nonalcoholic fatty liver disease activity score)
- Blood test
  - Glycemia, insulin, HbA1c, C-peptide
  - Liver function: total bilirubin, transaminases (ALAT, ASAT), gamma-glutamyltransferase (gammaGT)
  - Proteins and lipids: HDL, LDL, alpha2-macroglobulin, haptoglobin, apoA1, CrPus
  - Platelets, lymphocytes

Normalization

- Ordered quantile (ORQ) normalization

with the R-package `bestNormalize` (Ryan Andrew Peterson 2022; Ryan A. Peterson and Cavanaugh 2020)



Imputation

- K-nearest neighbors (kNN) with Gower's distance

Distance between observations i and j:

$$S_{ij} = \frac{\sum_{k=1}^{p} s_{ijk}\delta_{ijk}}{\sum_{k=1}^{p} \delta_{ijk}}$$

where $s_{ijk} \in [0,1]$ represents the similarity between i and j considering the variable k, $\delta_{i,j,k}$ indicates whether i and j can be compared along k.

For continuous variables:

$s_{ijk} = 1 - \frac{|x_{ik} - x_{jk}|}{r_k}$

with r_k the range of k.

For categorical variables:

$$s_{ijk} = \begin{cases} 0 & if\ x_{ik} = x_{jk}, \\ 1 & if\ x_{ik} \neq x_{jk} \end{cases}$$

Selection of phenotypes

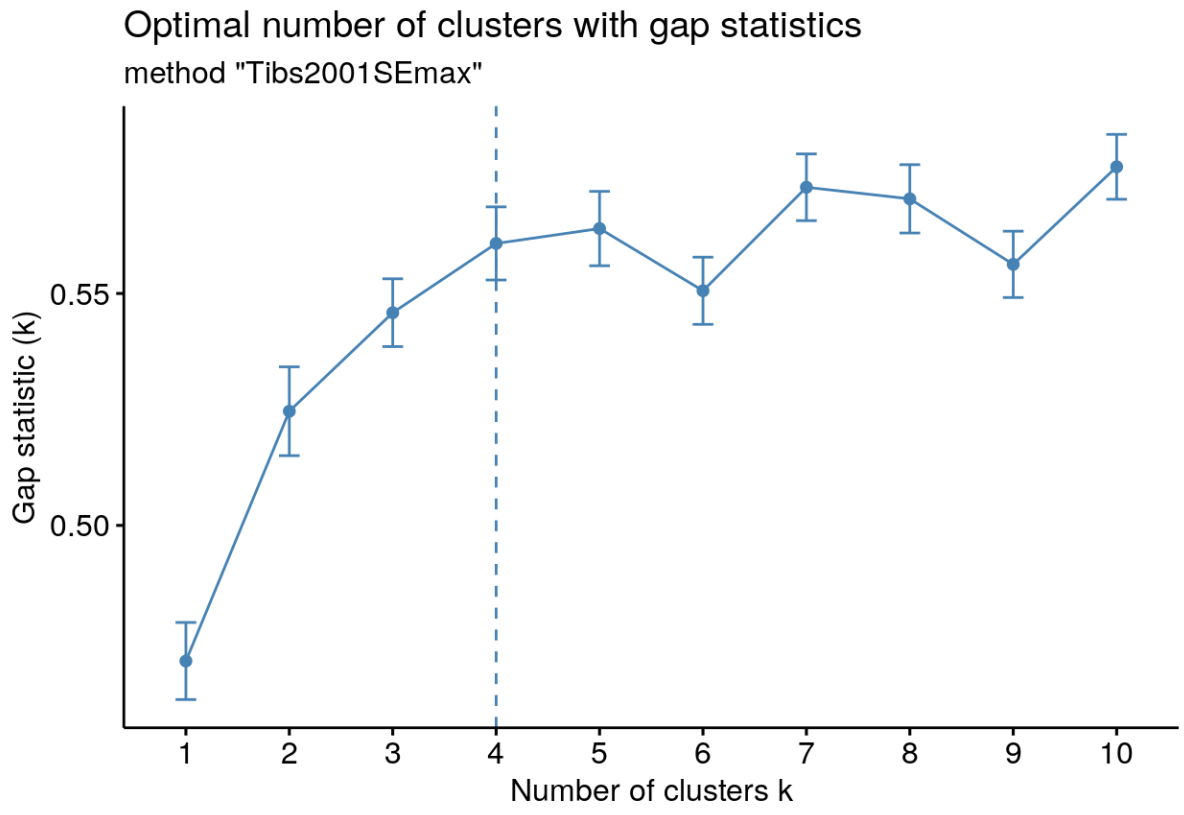
- Principal component analysis (PCA)
  - Use the elbow method to get the first n components to consider
  - Screen variables which contribute more than the average of the first n components

$\rightarrow$ Keep 13 out of 35 phenotypes.

K-means clustering

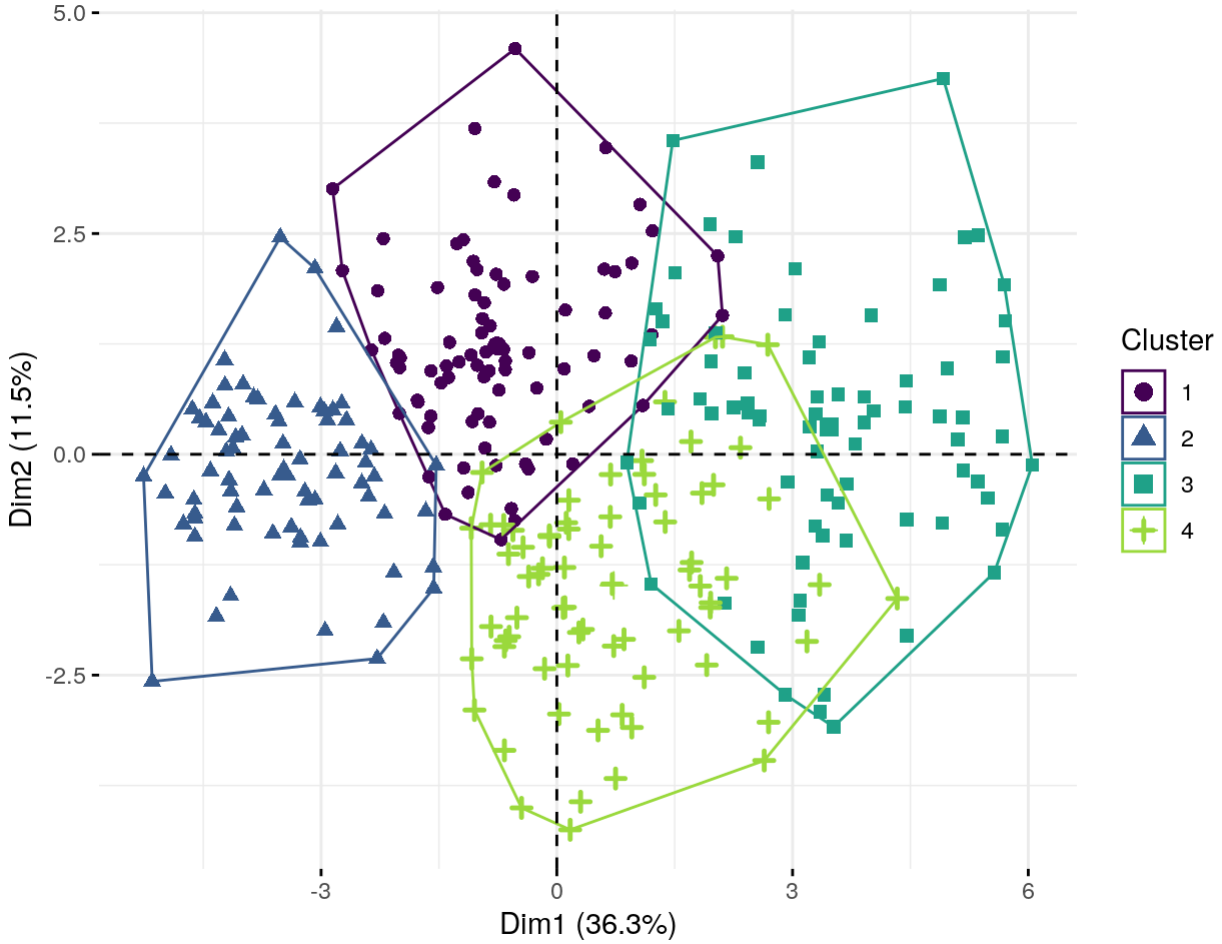
- The optimal number of clusters?

Methods: Elbow method, Silhouette coefficient, Gap statistic, …

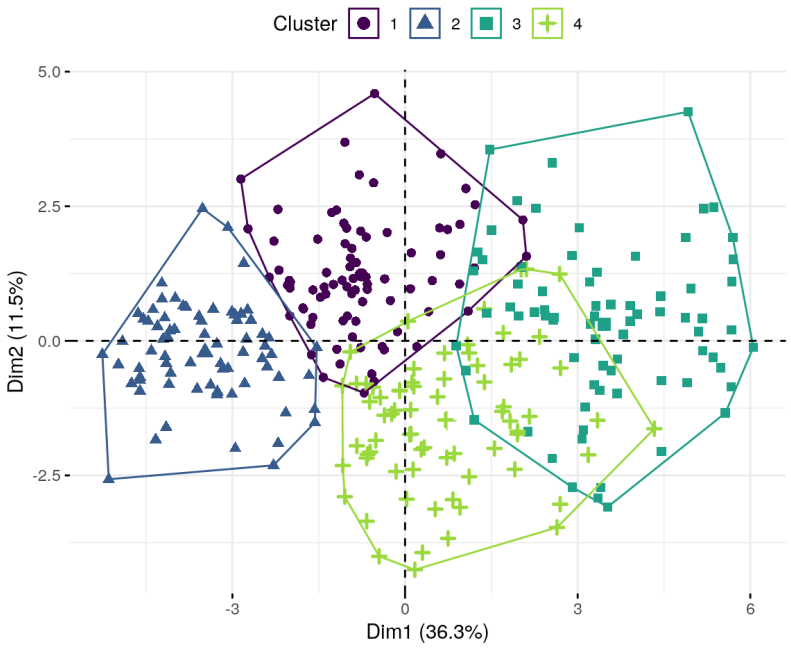


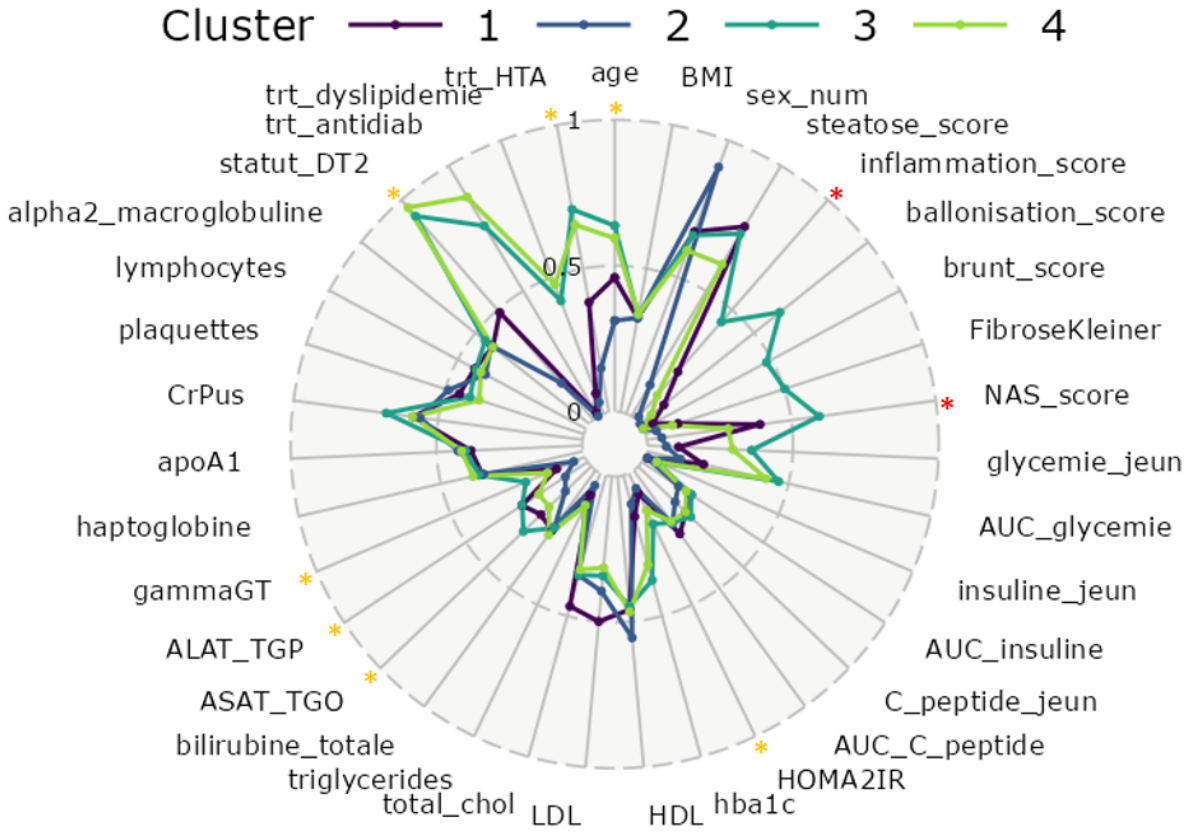
K-means clustering

$\rightarrow$ best $k = 4$



Phenotype-based clusters

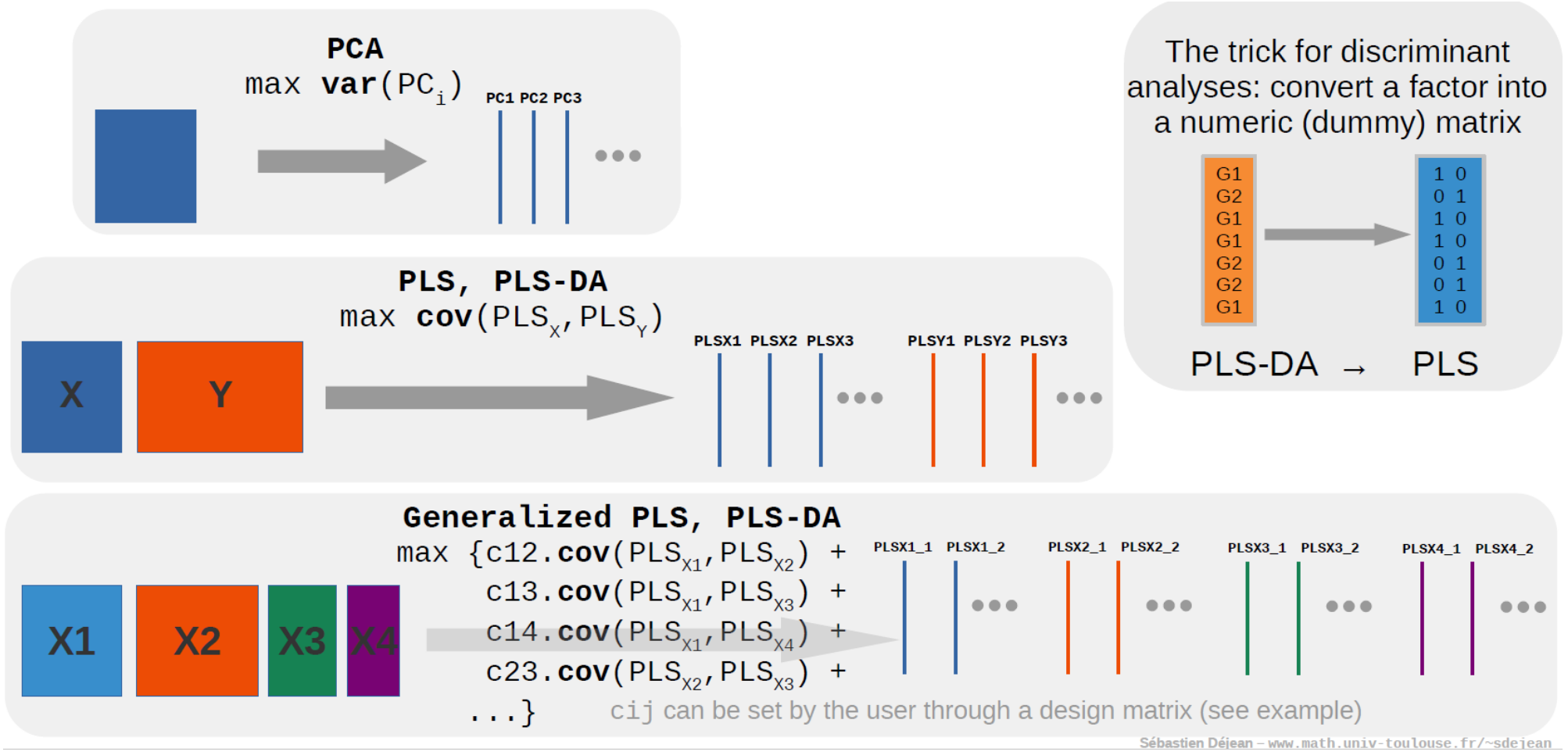




Means' comparison by Tukey HSD method:
* different for all pairs of groups;
* different for 5 pairs of groups.

Multi-omics analysis

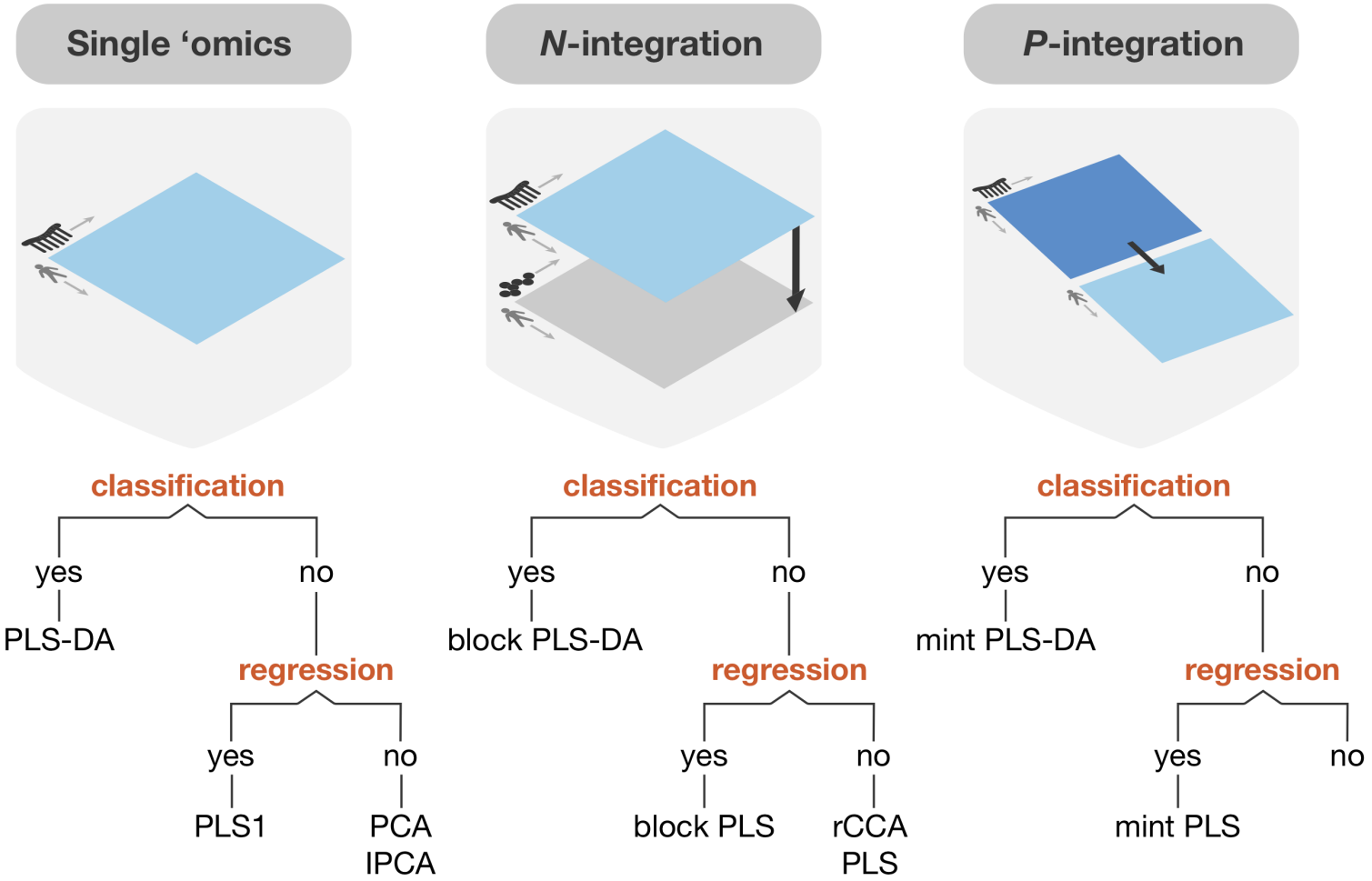
Integration problem



Source: extract from the presentation of Sébastien Déjean at the RIS thematic day at 2020.

R-package mixOmics

Types of analysis with `mixOmics` (Le Cao et al. 2021; F et al. 2017):



Source: schema from mixOmics' Hands-on activities.

N-integration

→ block PLS-DA (the DIABLO framework)

- Principle: latent components are being constructed such that the sum of covariances between all pairs of datasets is maximized and meanwhile discriminating the sample groups.

- Main function `block.splsda`
  - `block`: two or more datasets
  - `s` (sparse), method for feature selection, use cross-validation to tune:
    - number of components to use
    - number of features to keep
  - `pls`: partial least square regression
  - `da`: discriminant analysis (supervised method)

Principal steps (1)

Build the basic model:

```r
# library(mixOmics)

omics_list <- list(
  "PGS" = pgs_data,        # polygenetic scores
  "mehtyl" = methyl_data,
  "rnaseq" = rnaseq_data
)

design <- matrix(
  0.1,
  ncol = length(omics_list),
  nrow = length(omics_list),
  dimnames = list(names(omics_list), names(omics_list))
)
diag(design) <- 0

basic_diablo <- block.splsda(
  X = omics_list,          # list of data matrices
  Y = clusters,            # categorical outcome
  scale = TRUE,
  design = design,         # correlation matrix between dataset
  ncomp = 5                # an arbitrarily high number
)
```

Principal steps (2)

Tune parameters:

```r
## Number of components to keep ----
perf_diablo <- perf(
  object = basic_diablo,
  validation = "Mfold",
  folds = 10,
  nrepeat = 10
)
ncomp <- perf_diablo$choice.ncomp$WeightedVote["Overall.BER", "centroids.dist"]

## Number of features to keep ----
test_keepX <- list(
  "PGS" = ceiling(seq(5, ncol(pgs_data), length = 20)),
  "methylation" = ceiling(seq(5, ncol(methyl_data), length = 20)),
  "RNAseq" = ceiling(seq(5, ncol(rnaseq_data), length = 20))
)
tune_diablo <- tune.block.splsda(
  X = omics_list, Y = clusters,
  ncomp = ncomp, test.keepX = test_keepX, design = design,
  validation = "Mfold", folds = 3, nrepeat = 5,
  dist = "centroids.dist"
)
list_keepX <- tune_diablo$choice.keepX
```

Principal steps (3)

Build final model:

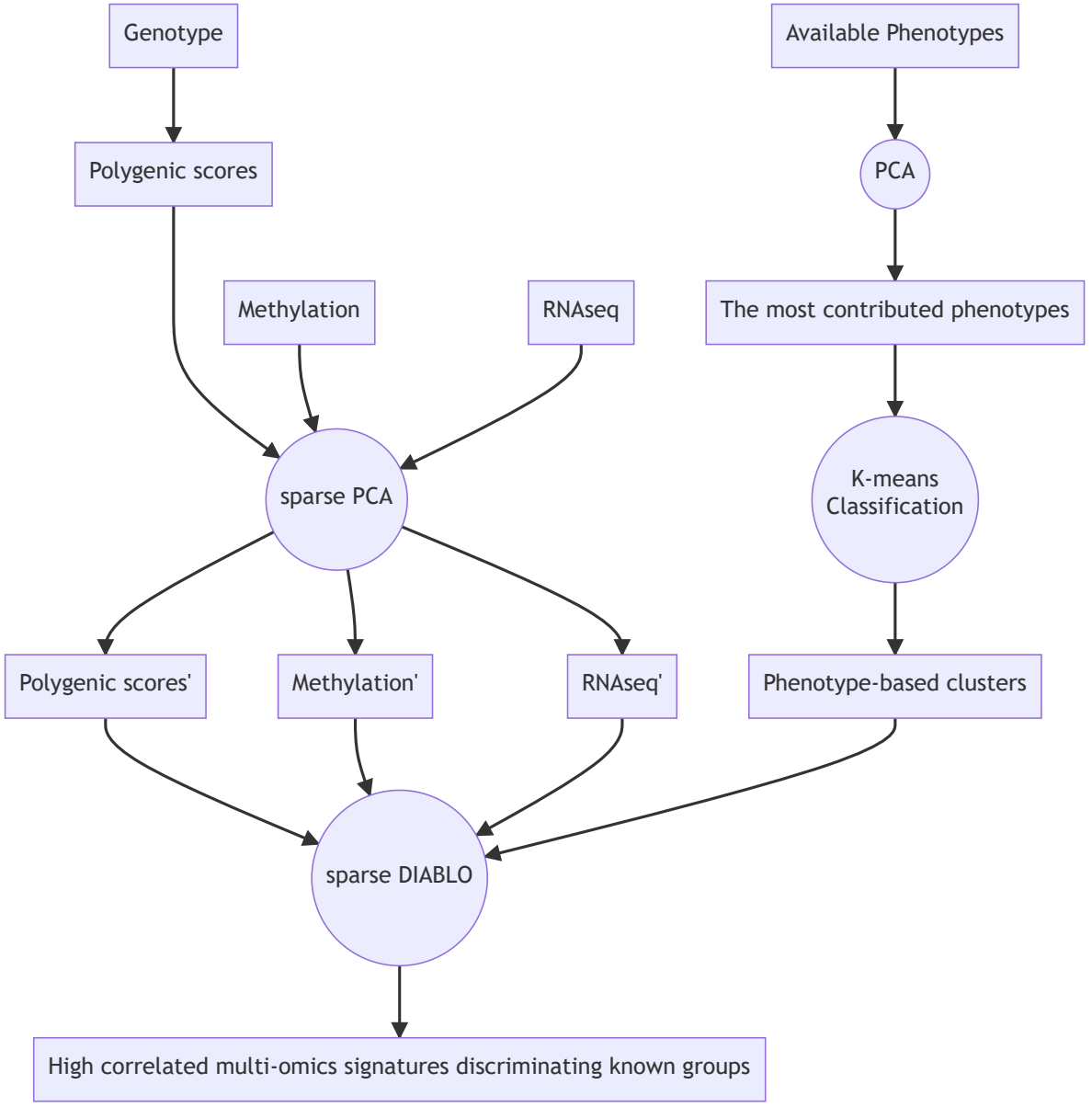
```
final_diablo <- block.splsda(
  X = omics_list,
  Y = clusters,
  ncomp = ncomp,
  keepX = list_keepX,
  design = design
)
```

Predict new samples:

```
pred_diablo <- predict(
  object = final_diablo,
  newdata = list(                # allow missing blocks, but not missing features
    "PGS" = new_pgs_data,
    "methyl" = new_methyl_data,
    "rnaseq" = new_rnaseq_data
  )
)

get.confusion_matrix(
  truth = new_sample_cluster,
  predicted = pred_diablo$WeightedVote$centroids.dist[, 2]
)
```

Workflow overview



- Number of features
  - Begin: 84 PGS, ≈ 740k CpGs, ≈ 14k genes
  - sPCA preselection: 44 PGS, 800 CpGs, 180 genes
  - Final selected: 24 PGS, 800 CpGs, 29 genes
- 40 hours + 108 CPUs + max 200Gb

Example of results

Selected features

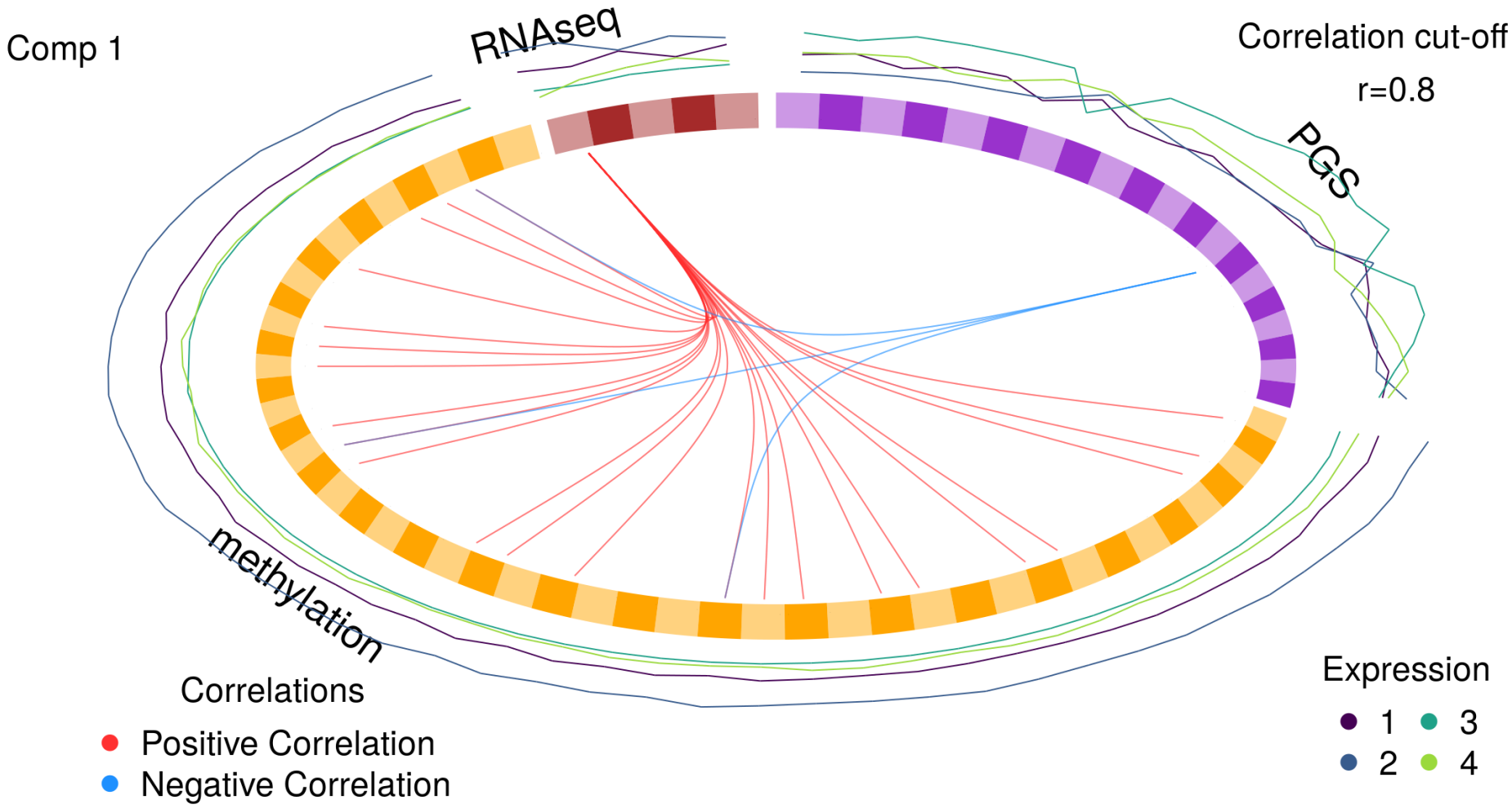
With the two first components, total features selected:

- 24 PGS
- 800 CpGs
- 29 genes (RNAseq)

Features selected in the first component:

- 20 PGS
- 47 CpGs
- 5 genes (RNAseq)

Correlation between features



Selected features

Features selected by the first component:

- 20 PGS
- 47 CpGs
- 5 genes (RNAseq)

Features that highly correlate (r > 0.8) with multiple features from other blocks:

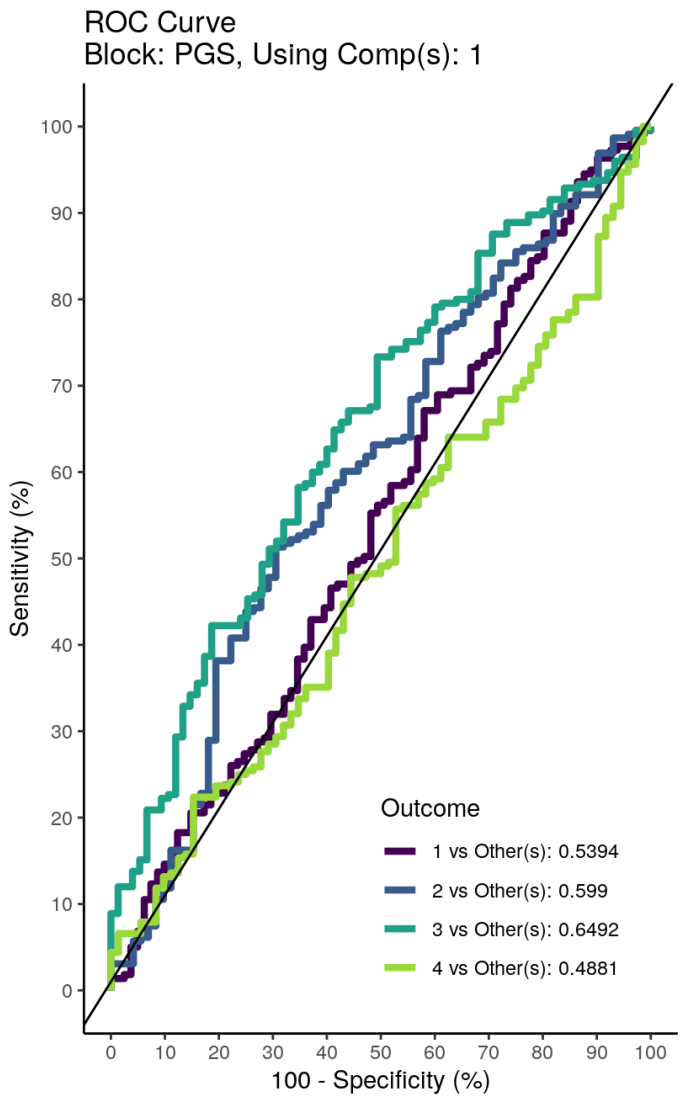
- ENSG00000111700.13 (*SLCO1B3*)

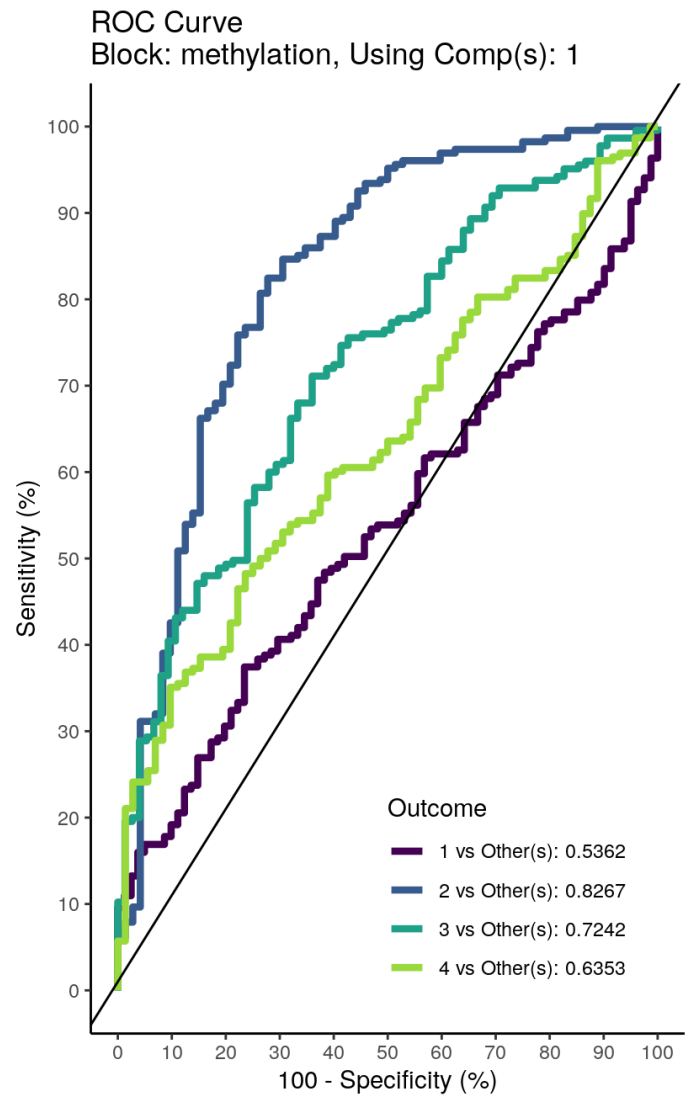
  This gene encodes a liver cell specific protein (OATP1B3), which helps to clear compounds from the body.

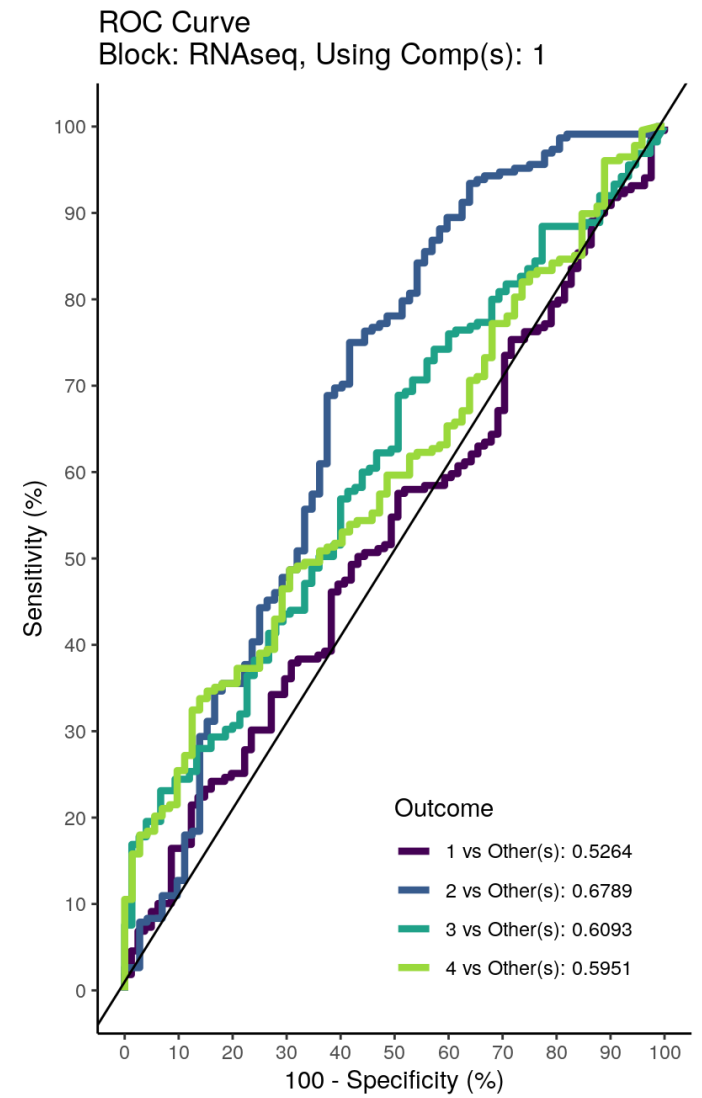
- PGS002308

  A trans-ancestry PGS built for the type 2 diabetes. (Ge, T., Irvin, M.R., Patki, A. *et al*. Genome Med (2022), doi: 10.1186/s13073-022-01074-2)

Area under the ROC







What more?

- Explore in depth the high correlated features & Refine parameter tuning
- Test to predict group for new samples

- There are more solutions than problems 💡

Thank you for your attention!

Any questions?

References

F, Rohart, Gautier B, Singh A, and Le Cao K-A. 2017. “mixOmics: An r Package for ’Omics Feature Selection and Multiple Data Integration.” *PLoS Computational Biology* 13 (11): e1005752. http://www.mixOmics.org.

Le Cao, Kim-Anh, Florian Rohart, Ignacio Gonzalez, and Sebastien Dejean. 2021. *mixOmics: Omics Data Integration Project*. http://www.mixOmics.org.

Peterson, Ryan A., and Joseph E. Cavanaugh. 2020. “Ordered Quantile Normalization: A Semiparametric Transformation Built for the Cross-Validation Era.” *Journal of Applied Statistics* 47 (13-15): 2312–27. https://doi.org/10.1080/02664763.2019.1630372.

Peterson, Ryan Andrew. 2022. *bestNormalize: Normalizing Transformation Functions*. https://CRAN.R-project.org/package=bestNormalize.