MOFA
Multi-Omic Factor Analysis

Method | 20 June 2018 | OPEN ACCESS

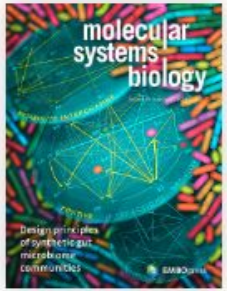
TRANSPARENT PROCESS

Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets

Ricard Argelaguet, Britta Velten, Damien Arnol, Sascha Dietrich, Thorsten Zenz, John C Marioni, Florian Buettner, Wolfgang Huber, Oliver Stegle

Author Information

Molecular Systems Biology (2018) 14: e8124 | https://doi.org/10.15252/msb.20178124

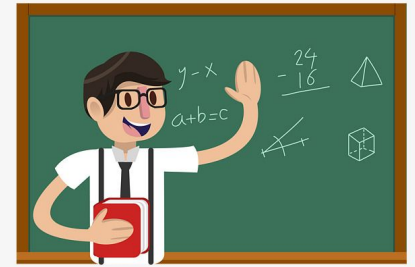


About the cover

Volume 14
Issue 6
1 June 2018

IN THIS ISSUE

Matrix multiplication, back to school



$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 10 & 11 \\ 20 & 21 \\ 30 & 31 \end{bmatrix}$$

$$= \begin{bmatrix} 1x10 + 2x20 + 3x30 & 1x11 + 2x21 + 3x31 \\ 4x10 + 5x20 + 6x30 & 4x11 + 5x21 + 6x31 \end{bmatrix}$$

$$= \begin{bmatrix} 10+40+90 & 11+42+93 \\ 40+100+180 & 44+105+186 \end{bmatrix} = \begin{bmatrix} 140 & 146 \\ 320 & 335 \end{bmatrix}$$

$$c_{11} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31} + a_{14}b_{41}$$

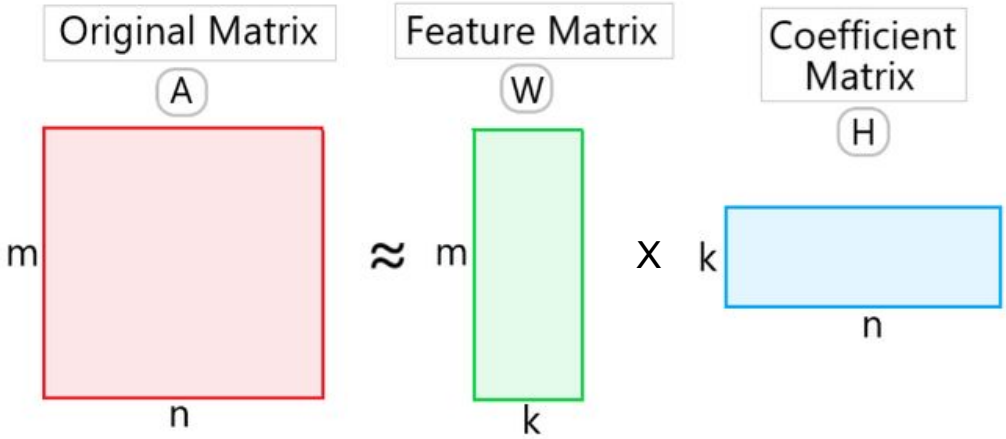
$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{bmatrix}$$

Matrix dimensions : 2 x 4 ..... 4 x 3 2 x 3

$$c_{22} = a_{21}b_{12} + a_{22}b_{22} + a_{23}b_{32} + a_{24}b_{42}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \\ b_{41} & b_{42} & b_{43} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \end{bmatrix}$$

Matrix factorization



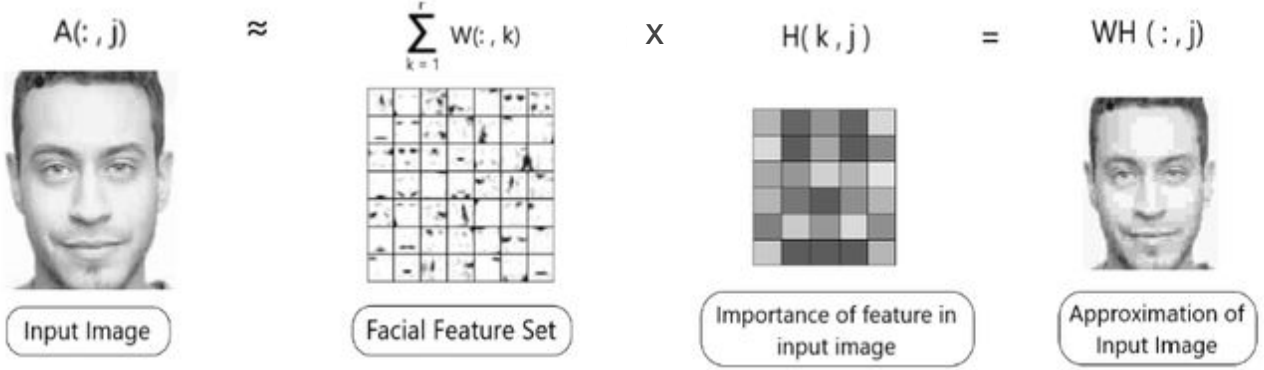
→ approximate the large data matrix A using the product of 2 smaller matrices W and H

$$A = W \times H + \varepsilon$$

Matrix factorization applications

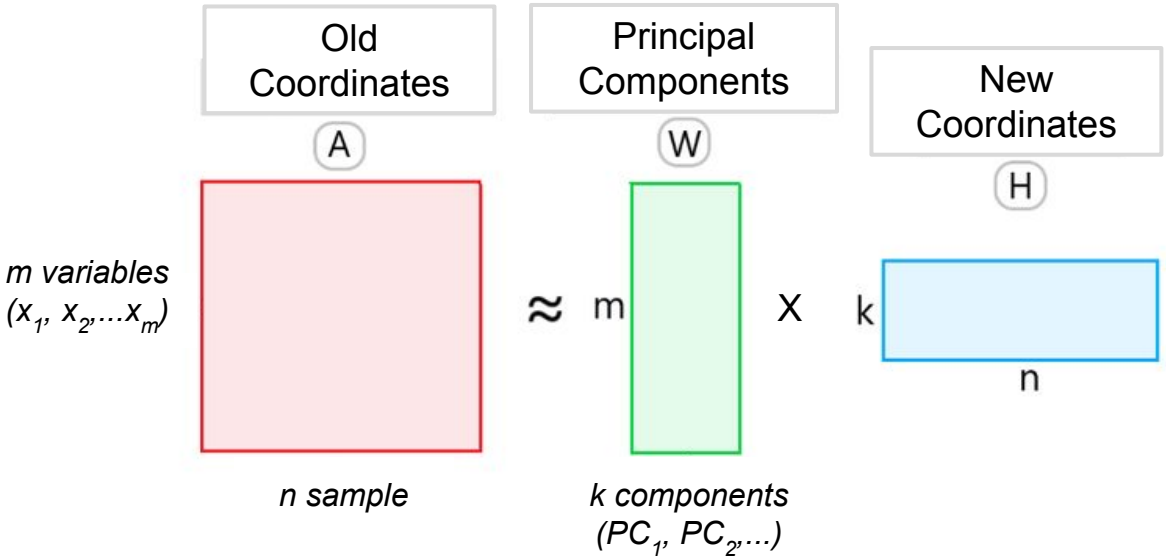
Useful for dimensionality reduction (k features) and feature extraction (the H matrix)

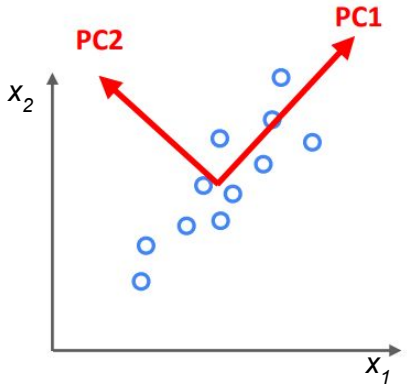
Example: image processing with Non-Negative Matrix Factorization ($W \geq 0$ and $H \geq 0$)



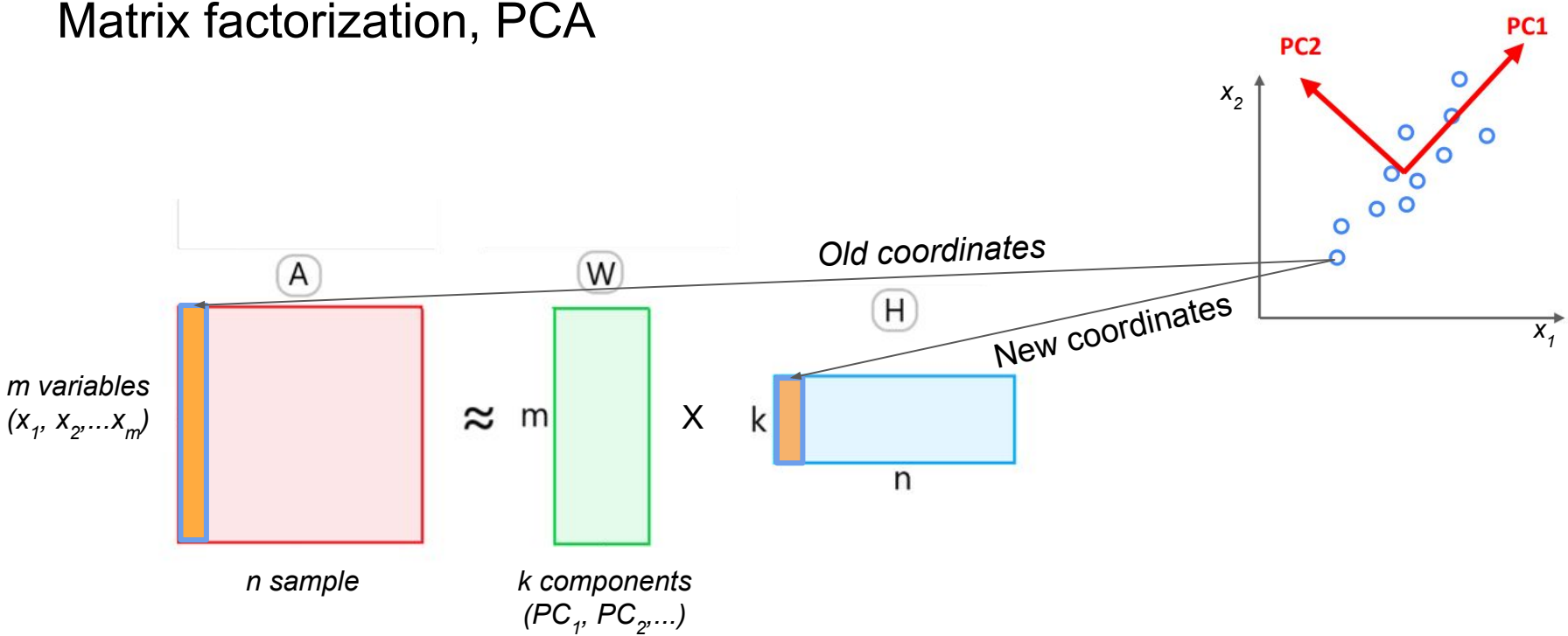
→ PCA can be formulated as an approximation of matrix factorization

Matrix factorization, PCA

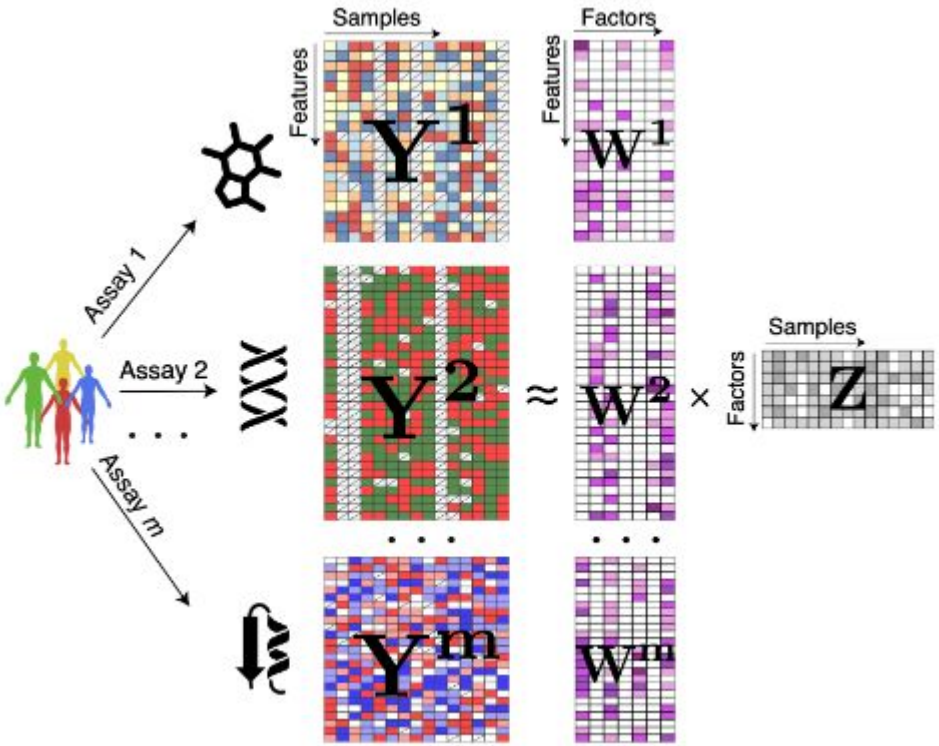




Matrix factorization, PCA



MOFA : PCA generalization



$$\mathbf{Y}^m = \mathbf{Z}\mathbf{W}^{mT} + \boldsymbol{\epsilon}^m$$

- *m* views for *m* omic sources
- share the Z matrix between views
- 2 levels of sparsity on W^m :
  - view and factor-wise
    → active/inactive factors in *a view*
  - feature-wise
    → sparse biological phenomenon
- Y^m and ε^m can follow different models :
  - <u>Gaussian (continuous)</u>
  - Poisson (natural/count)
  - Bernouilli (binary)

MOFA : How it works ?

$$\mathbf{Y}^m = \mathbf{Z}\mathbf{W}^{mT} + \boldsymbol{\epsilon}^m$$

- Parameter estimation through variational Bayesian inference

$$p(\mathbf{Y}, \hat{\mathbf{W}}, \mathbf{S}, \mathbf{Z}, \boldsymbol{\Theta}, \boldsymbol{\alpha}, \boldsymbol{\tau}) = \prod_{m=1}^{M}\prod_{n=1}^{N}\prod_{d=1}^{D_m}\mathcal{N}\left(y_{nd}^m \,|\, \sum_{k=1}^{K} s_{dk}^m \hat{w}_{dk}^m z_{nk}, 1/\tau_d\right)$$
$$\prod_{m=1}^{M}\prod_{d=1}^{D_m}\prod_{k=1}^{K}\mathcal{N}\left(\hat{w}_{dk}^m \,|\, 0, 1/\alpha_k^m\right)\mathrm{Ber}(s_{d,k}^m|\theta_k^m)$$
$$\prod_{n=1}^{N}\prod_{k=1}^{K}\mathcal{N}\left(z_{nk} \,|\, 0, 1\right)$$
$$\prod_{m=1}^{M}\prod_{k=1}^{K}\mathrm{Beta}\left(\theta_k^m \,|\, a_0^\theta, b_0^\theta\right)$$
$$\prod_{m=1}^{M}\prod_{k=1}^{K}\mathcal{G}\left(\alpha_k^m \,|\, a_0^\alpha, b_0^\alpha\right)$$
$$\prod_{m=1}^{M}\prod_{d=1}^{D_m}\mathcal{G}\left(\tau_d^m \,|\, a_0^\tau, b_0^\tau\right).$$

posterior distribution of unobserved data X, P(X/Y), is approximated by $q(\mathbf{X}) = \prod_i q(\mathbf{X}_i)$

$$q(\mathbf{Z}, \mathbf{S}, \hat{\mathbf{W}}, \boldsymbol{\alpha}, \boldsymbol{\tau}, \boldsymbol{\theta}) = q(\mathbf{Z})q(\boldsymbol{\alpha})q(\boldsymbol{\theta})q(\boldsymbol{\tau})q(\mathbf{S}, \hat{\mathbf{W}})$$

MOFA : How it works ?

$$\mathbf{Y}^m = \mathbf{Z}\mathbf{W}^{mT} + \boldsymbol{\epsilon}^m$$

- Parameter estimation through variational Bayesian inference
- Evidence Lower Bound (ELBO)
  - the true log marginal likelihood $\log p(\mathbf{Y})$ is lower bound by the ELBO $\mathcal{L}(\mathbf{X})$

$$\begin{aligned}\mathcal{L}(\mathbf{X}) &= \int q(\mathbf{X}) \Big(\log \frac{p(\mathbf{X}|\mathbf{Y})}{q(\mathbf{X})} + \log p(\mathbf{Y}) \Big) d\mathbf{X} \\ &= \log p(\mathbf{Y}) - \mathrm{KL}(q(\mathbf{X}) || p(\mathbf{X}|\mathbf{Y})) \\ &\leq \log p(\mathbf{Y})\end{aligned}$$

  - the objective is to optimise $\mathcal{L}(\mathbf{X})$ with respect to the distribution q(X)

MOFA : How it works ?

$$\mathbf{Y}^m = \mathbf{Z}\mathbf{W}^{mT} + \boldsymbol{\epsilon}^m$$

- Parameter estimation through variational Bayesian inference
- Evidence Lower Bound (ELBO)
- Iterative estimation process similar to the Expectation-Maximization (EM) algorithm
  - each unobserved variable is updated one by one considering the others

$$q(\mathbf{Z}) = \prod_{k=1}^{K}\prod_{n=1}^{N} q(z_{nk}) = \prod_{k=1}^{K}\prod_{n=1}^{N} \mathcal{N}\left(z_{nk} \,|\, \mu_{z_{nk}}, \sigma_{z_{nk}}\right)$$

$$q(\hat{\mathbf{W}}, \mathbf{S}) = \prod_{m=1}^{M}\prod_{d=1}^{D_m}\prod_{k=1}^{K} q(\hat{w}_{dk}^m, s_{dk}^m) = \prod_{m=1}^{M}\prod_{d=1}^{D_m}\prod_{k=1}^{K} q(\hat{w}_{dk}^m | s_{dk}^m) q(s_{dk}^m)$$

MOFA : How it works ?

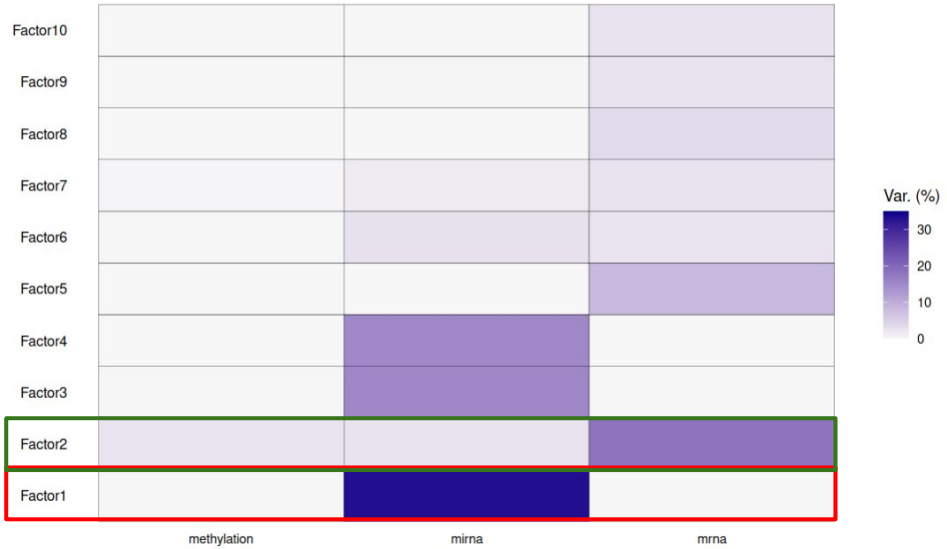
$$\mathbf{Y}^m = \mathbf{Z}\mathbf{W}^{mT} + \boldsymbol{\epsilon}^m$$

- Parameter estimation through variational Bayesian inference
- Evidence Lower Bound (ELBO)
- Iterative estimation process similar to the Expectation-Maximization (EM) algorithm
- Iteration stop when ELBO change is small enough
- Automatically drop factors with low variance explained…

MOFA results

Variance decomposition by factors

→ percentage of variance explains by each factor across each data modality

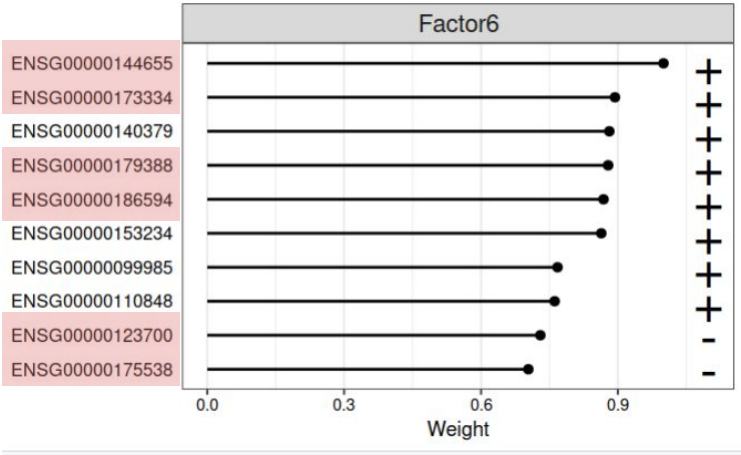


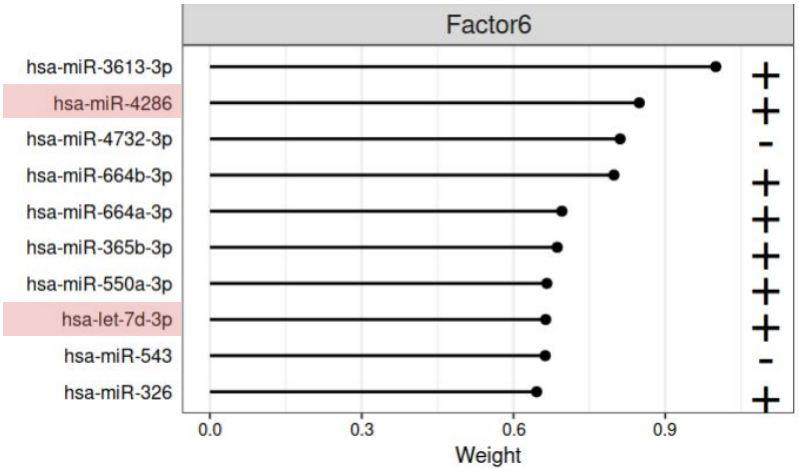
- Factor 1 captures a source of variability that is present mainly in the miRNA view
- Factor 2 captures variation that is present across all data modalities but mainly in mRNA.

MOFA results on W matrices

Feature weights by factor for each view/omic (ie a W^m column)

→ weights provide a signed score (association measure) for each feature for a given factor (below mRNA and miRNA for Factor 6, associated with Sample_Group variable)



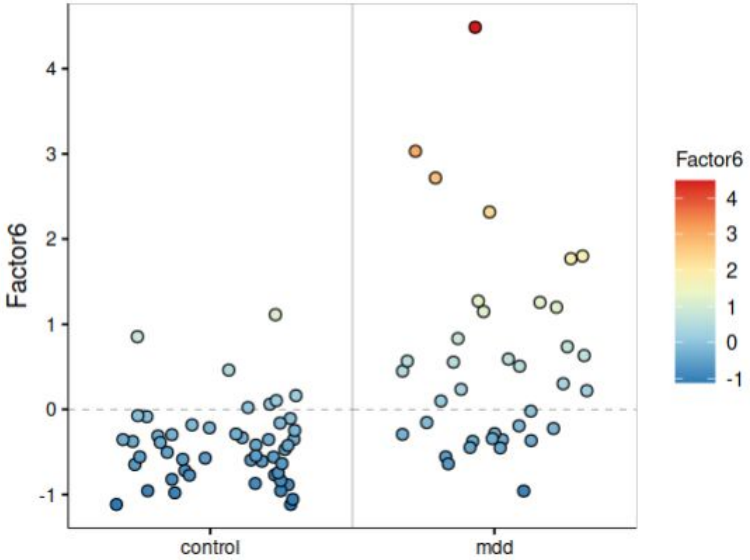


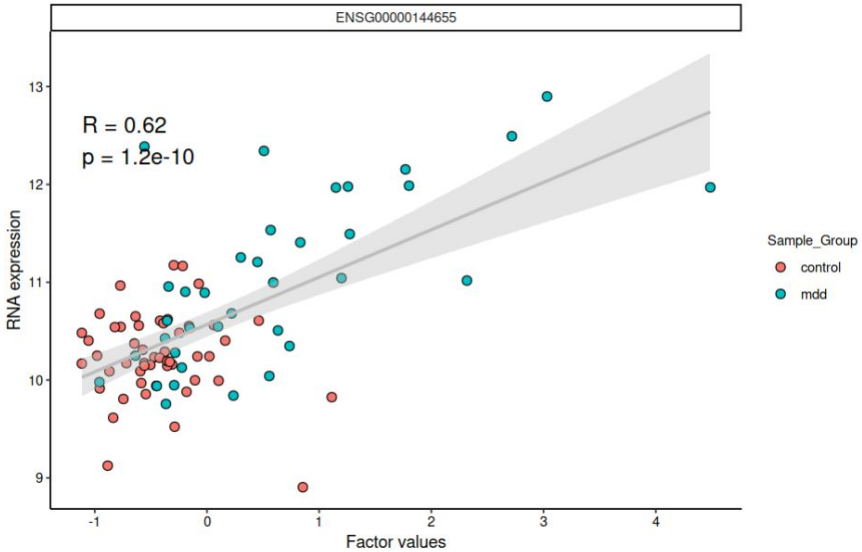
Identified as significant in Component 1 of supervised RGCCA (the most associated to Sample_Group variable)

MOFA results on Z matrix

Factor values regarding known groups of samples (ie a Z row)

→ detect association between a factor and a specific variable/feature





- Separation between control and MDD patients shows association with Factor 6
- Expression of CSRNP1 gene (ENSG00000144655) is also associated with Factor 6 (and MDD status)

MOFA characteristics

- Choice of **k** (number of factors)
  → inactive factors can be removed automatically during learning (or through a user defined explained variance threshold)

- Random initialization : no guarantee of optimal solution during estimation
  → run MOFA several times (~10 times) with different initialisations (solved in MOFA+)
  → keep the model with the highest ELBO for downstream analysis

- Missing value
  → no need for imputation, missing values are ignored in the model thanks to probabilistic modelling

- Data pre-processing
  → no need as long as indicated distributions are respected (eg. Gaussian) → to check ++

MOFA limits

- Differents views but on the same sample
- Mainly linear relationships are captured
- Assumes independence between features
- Unbalanced groups sensibility