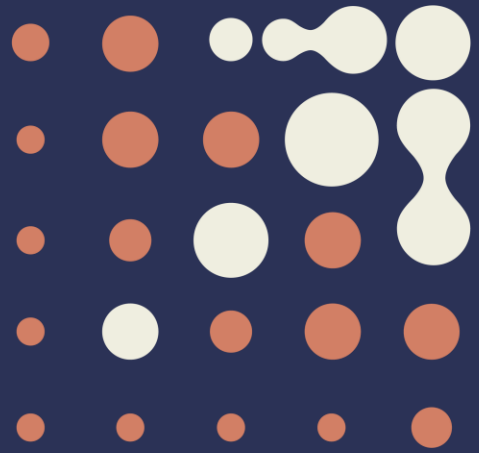
Medical Genomics Practical #2: Multi-omic analyses in R

N. Alcala

Rare Cancers Genomics Team

November 23th & 30th 2022



International Agency for Research on Cancer

World Health Organization

RARE CANCERS GENOMICS

Plan

Practical (3.5 hrs)

- **Concepts**: interfacing R and python, reminder of MOFA model, MOFA2 implementation
- **Practical**: unsupervised analyses of rare lung cancers

Projects (0.5 hrs)

- **Choice**
- **Quickstart**

Concepts | *Interfacing R and python*

Why choose between python and R?

R Vs Python: What's the Difference?

R and Python are both open-source programming languages with a large community. New libraries or tools are added continuously to their respective catalog. R is mainly used for statistical analysis while Python provides a more general approach to data science.

R and Python are state of the art in terms of programming language oriented towards data science. Learning both of them is, of course, the ideal solution. R and Python requires a time-investment, and such luxury is not available for everyone. Python is a general-purpose language with a readable syntax. R, however, is built by statisticians and encompasses their specific language.

In this tutorial, you will learn

- R
- Python
- Popularity index
- Job Opportunity
- Analysis done by R and Python
- Percentage of people switching
- Difference between R and Python
- R or Python Usage

Python vs R for Data Science: And the winner is..

Data-Driven Science Jan 31, 2018 · 8 min read

About: Data-Driven Science (DDS) provides training for people building a career in Artificial Intelligence (AI). Follow us on Twitter.

R vs Python for Data Analysis — An Objective Comparison

Concepts | *Interfacing R and python*

Why choose between python and R?

- The **core of the MOFA method is in python** (mofapy, mofapy2) and uses the powerful python machine learning packages (e.g., scikit-learn)
- The **downstream analyses** and **graphical functions are in R** and leverage the contributions of the enormous R community of computational biologists and statisticians

Issues

- Need to correctly **interface python and R**
- **Many dependencies** in both languages, making the code **difficult to set up and fragile**

Concepts | *Interfacing R and python*

Why choose between python and R?

- The **core of the MOFA method is in python** (mofapy, mofapy2) and uses the powerful python machine learning packages (e.g., scikit-learn)
- The **downstream analyses** and **graphical functions are in R** and leverage the contributions of the enormous R community of computational biologists and statisticians

Solutions

1. (earlier MOFA and MOFA+ releases) **R package reticulate:** allows to specify a python install or conda env, run python functions, transfer R and Pandas data frames, or R matrices and NumPy arrays
2. (newer MOFA+ releases) **R bioconductor package basilisk**: allows R to directly create and handle conda environments with specific python dependencies, allowing smooth usage of incompatible python installs within a same R session

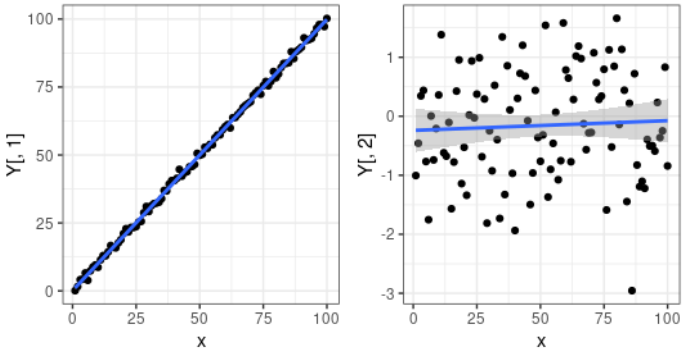
Concepts | *Multivariate regression reminder*

Multivariate linear models

- model **Y = AX + ε**,
- where **Y** is the matrix of observations for *M* features (rows) and *N* sample (columns)
- **X** is the known covariable matrix (*K* by *N*)
- **A** is the unknown weights matrix (*M* by *K*)
- **ε** is the residual noise (*M* by *N*)

Example: effect of smoking on gene expression in 100 patients

- Y: 2 genes x 100 patients
- X: smoking in pack-years for 100 patients
- A=[1,0]^t



Example MLM. The expression of the first gene (Y[1,]) is associated with smoking (x). The expression of the second gene (Y[2,]) is independent of (x)

Concepts | *Multivariate regression reminder*

Principal Component Analysis

- model **Y = WZ + ε**,
- where **Y** is the matrix of observations for *M* features (rows) and *N* samples (columns)
- **Z** is the unknown latent variable matrix (*K* by *N*)
- **W** is the unknown weights matrix (*M* by *K*)
- **ε** is the residual noise (*M* by *N*)

=> similar to regression but we do not have access to the "true covariable" explaining the variation in Y, we try to guess

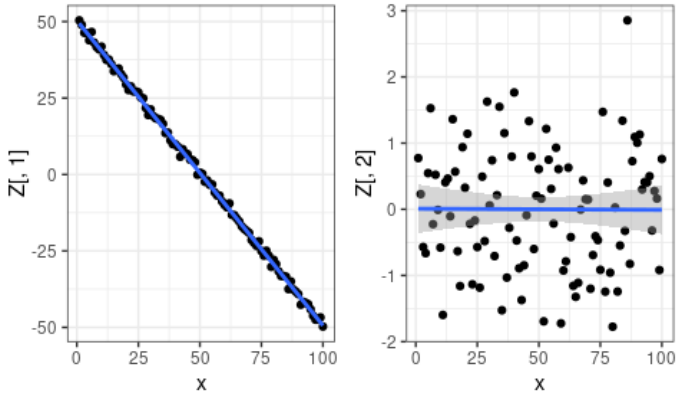
Concepts | *Multivariate regression reminder*

Principal Component Analysis

- model **Y = WZ + ε**,
- where **Y** is the matrix of observations for *M* features (rows) and *N* samples (columns)
- **Z** is the unknown latent variable matrix (*K* by *N*)
- **W** is the unknown weights matrix (*M* by *K*)
- **ε** is the residual noise (*M* by *N*)

Example: unknown effect of smoking on gene expression in 100 patients

- Y: 2 genes x 100 patients
- X: smoking in pack-years for 100 patients is unknown
- Z: 2 latent variables x 100 patients, obtained by the PCA
- W=[-1,0]^t, direction is not important

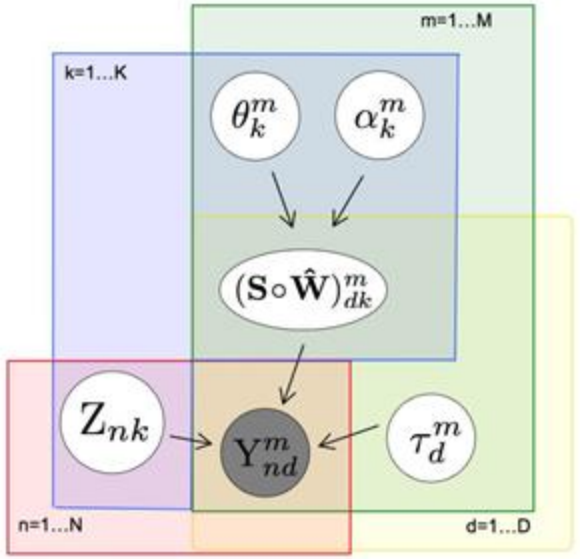


Example PCA. The first latent variable (Z[,1]) recapitulates smoking (x) although the model was blind to x, because the first gene (Y[,1]) is highly variable and strongly correlated with x **=> the PCA "found" the hidden covariable x**

Concepts | *MOFA reminder*

Multi-Omics Factor Analysis (MOFA)

- **Generalization of Principal Component Analysis to multiple modalities** *M*
- model $\mathbf{Y}^m = \mathbf{W}^m\mathbf{Z} + \boldsymbol{\varepsilon}^m$,
- where $\mathbf{Y}^m$ is the matrix of observations for each feature *d* (rows) and each sample *n* (columns) for modality *m* (e.g., genomic alterations, expression)
- **Z** is the latent factors matrix (*K* by *N*) shared by all modalities *m*
- $\mathbf{W}^m$ is the weights (loadings) matrix (M by *K*) of *m*
- $\boldsymbol{\varepsilon}^m$ is the residual noise (*M* by *N*)



MOFA directed acyclic graph. *Source: Argelaguet et al. Mol Syst Biol 2018.*

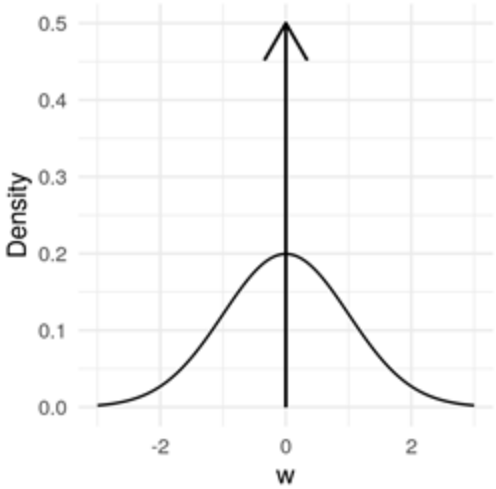
Concepts | *MOFA implementation*

Multi-Omics Factor Analysis (MOFA)

- Model $\mathbf{Y}^m = \mathbf{W}^m\mathbf{Z} + \boldsymbol{\varepsilon}^m$

Bayesian inference of elements of $\mathbf{Z}$ and $\mathbf{W}^m$

- *Sparse* (Automatic Relevance Determination X "spike-and-slab") priors on weights $w^m_{d,k} = s^m_{d,k}\hat{w}^m_{d,k}$, with priors $s^m_{d,k} \sim$ Bernoulli(Θ^m_k) and $\hat{w}^m_{d,k} \sim$ Normal(0,1/α^m_k), **so in modality *m*, if Θ^m_k is close to 0, factor *k* is sparse (most features have 0 weights), and if α^m_k is large factor k not active (e.g., the factor does not explain any variation)**



Spike and slab prior. The arrow represents a Dirac point mass at 0.

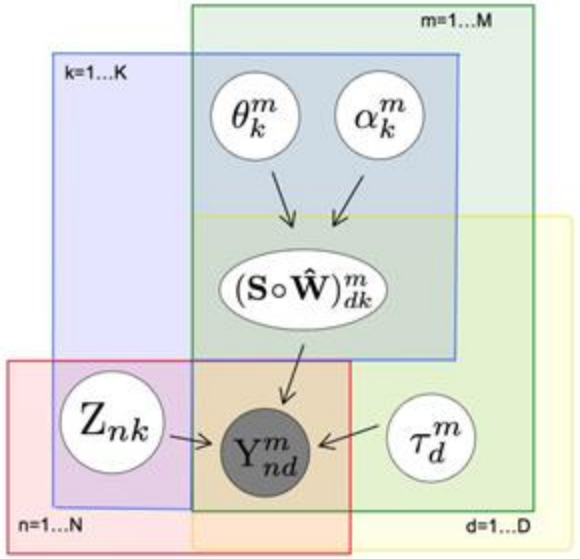
Concepts | *MOFA implementation*

Multi-Omics Factor Analysis (MOFA)

- Model $\mathbf{Y}^m = \mathbf{W}^m\mathbf{Z} + \boldsymbol{\varepsilon}^m$

Bayesian inference of elements of **Z** and $\mathbf{W}^m$

- *Gaussian* (for continuous data, e.g. normalized expression data and methylation *M* values), *Bernoulli* (for binary data, e.g. genomic alterations), or *Poisson* (for count data, e.g. as expression in read counts) **prior distributions on noise** $\varepsilon^m{}_n$



MOFA directed acyclic graph. *Source: Argelaguet et al. Mol Syst Biol 2018.*

Concepts | *MOFA implementation*

Multi-Omics Factor Analysis (MOFA)

Variational Bayes (or VI) implementation:

- *Rationale*: when fitting complex Bayesian models, the posterior distribution of the parameters is often intractable; we **need an approximation**
- *Method (VI)*: a **lower bound on the model likelihood (the Evidence Lower Bound--ELBO) is optimized** (E-M algorithm), using a simpler factorized form for the posterior
- *Note*: less computer-intensive alternative to the popular Monte Carlo Markov Chains (MCMC)

Practical 2

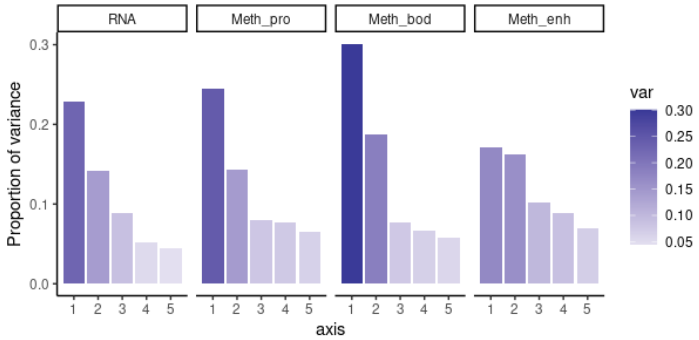
Using MOFA to build a molecular map of expression and methylation of the rare lung neuroendocrine neoplasms
https://github.com/IARCbioinfo/medical_genomics_course/wiki/Practical-2

- **preprocessing** with RNA-seq pipeline from Practical 1 and the DNA methylation array processing script at https://github.com/IARCbioinfo/Methylation_analysis_scripts
- **normalize expression data** with the VST function
- **convert methylation into M-values** (<0 for unmethylation, >0 for methylated)
- **split methylation data** into 3 datasets according to location (sites within gene promoter regions, within gene bodies, and within enhancers)
- **select the 5000 most variable features** for each of the 4 datasets

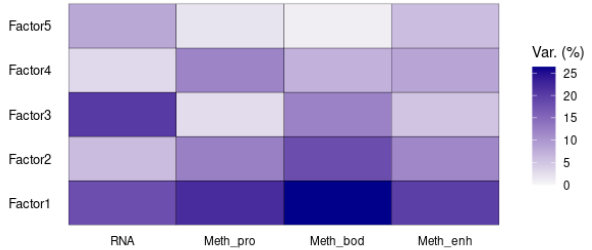
Understanding MOFA | *Link with PCA*

MOFA factors explain variance across the input matrices

- *Independent factorization with PCAs on each 'omic layer (input matrix):*
  - *Variance decreases monotonously with PC #*
  - *PCs do not necessarily match across PCAs*



- *Joint factorization with MOFA*
  - *Sum of variance decreases monotonously but not necessarily the variance of each 'omic layer*
  - *Factors are shared across 'omic layers*

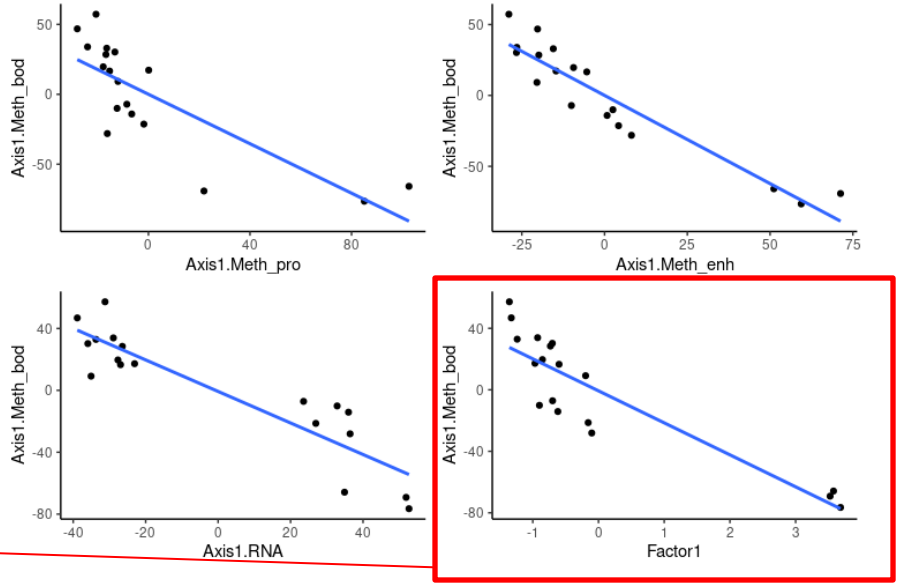


Understanding MOFA | *Link with PCA*

MOFA factors explain variance across the input matrices

- *Independent factorization with PCAs on each 'omic layer (input matrix):*
  - *Variance decreases monotonously with PC #*
  - *PCs do not necessarily match across PCAs*

- *Joint factorization with MOFA*
  - *Sum of variance decreases monotonously but not necessarily the variance of each 'omic layer*
  - *Factors are shared across 'omic layers and correspond to a compromise between PCs*

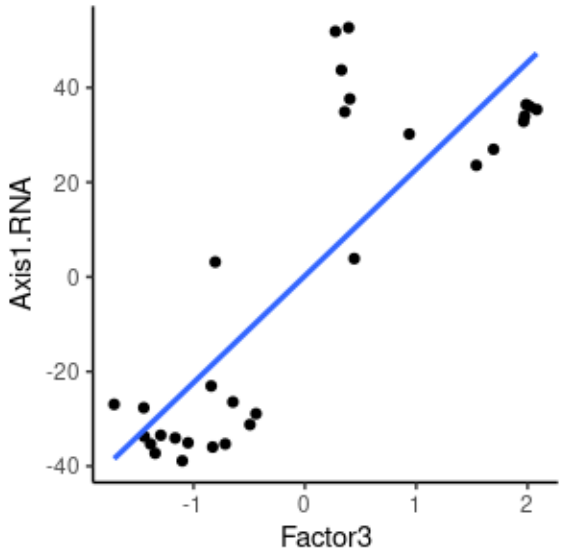


MOFA

Understanding MOFA | *Link with PCA*

MOFA factors explain variance across the input matrices

- *Independent factorization with PCAs on each 'omic layer (input matrix):*
  - *Variance decreases monotonously with PC #*
  - *PCs do not necessarily match across PCAs*



- *Joint factorization with MOFA*
  - *Sum of variance decreases monotonously but not necessarily the variance of each 'omic layer*
  - *Factors are shared across 'omic layers and correspond to a compromise between PCs, not necessarily in the same order*

Projects

Different flavors of computational biology for medical genomics

- I have heard of new analyses techniques that I want to try on my data => **Projects 1 and 2** (R scripting)
- I have new data that I want to explore and compare with previous data => **Project 3** (R scripting)
- I have scripts for a software that I want to implement in a reproducible workflow => **Project 4** (nextflow coding)
- I have data from new types/technologies and I want to explore their possibilities => **Project 5 and 6** (R scripting)